



UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO

Smart Parking: Uma Aplicação Para Estacionamento em Cidades Inteligentes

Trabalho de Conclusão de Curso

Samila Ruane Barboza Santos e Sílvia Rodrigo Lima Passos



São Cristóvão – Sergipe

2017

UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO

Samila Ruane Barboza Santos e Sílvia Rodrigo Lima Passos

Smart Parking: Uma Aplicação Para Estacionamento em Cidades Inteligentes

Trabalho de Conclusão de Curso submetido ao Departamento de Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Sistemas de Informação.

Orientador(a): Gilton José Ferreira da Silva

São Cristóvão – Sergipe

2017

Samila Ruane Barboza Santos e Sílvia Rodrigo Lima Passos

Smart Parking: Uma Aplicação Para Estacionamento em Cidades Inteligentes/ Samila Ruane Barboza Santos e Sílvia Rodrigo Lima Passos. – São Cristóvão – Sergipe, 2017-
188 p. : il. (algumas color.) ; 30 cm.

Orientador: Gilton José Ferreira da Silva

Trabalho de Conclusão de Curso – UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO, 2017.

1. Estacionamento inteligente. 2. Cidades inteligentes. 3. Aplicativos móveis. 4. Mobilidade Urbana I. Orientador. II. Universidade Federal de Sergipe. III. Título

CDU 02:141:005.7

Samila Ruane Barboza Santos e Sílvia Rodrigo Lima Passos

Smart Parking: Uma Aplicação Para Estacionamento em Cidades Inteligentes

Trabalho de Conclusão de Curso submetido ao Departamento de Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Sistemas de Informação.

Trabalho aprovado. São Cristóvão – Sergipe, 10 de Outubro de 2017:

Gilton José Ferreira da Silva
Orientador

Adicinea Aparecida De Oliveira
Convidado 1

Edward David Moreno
Convidado 2

São Cristóvão – Sergipe
2017

*Nós dedicamos este trabalho a Deus, a nossas famílias, amigos
e professores, os quais nos deram o suporte para chegarmos até aqui.*

Agradecimentos

Poderíamos dizer Obrigada!, *Thank you!*, *Merci!*, *Dank!*. Usar as 6909¹ línguas faladas no mundo não seria suficiente para agradecer a todo o suporte que recebemos da nossa família, amigos e professores. Seja nas madrugadas em claro estudando ou terminando algum projeto. Nos pães com queijo e mussarela nas reuniões de ESII, nos momentos zumbis derivados do corre e corre diário e da abdicação dos finais de semana. Só temos a agradecer de todo o coração.

É difícil citar todos, e aqueles que não estiverem aqui, perdoem a nossa má memória, mas pode ter certeza que somos muito gratos pelo seu suporte.

Primeiro, gostaríamos de agradecer a Deus, por ter nos sustentado nessa caminhada.

Um obrigado mais que especial para Claudio Mota Oliveira pela sua capacidade de tornar qualquer assunto, a primeira vista monstruoso, em algo simples e por estar disposto a ajudar sempre que precisamos.

As nossas mães, Lucivane Barboza e Maria Passos pelo amor incondicional que nos manteve firme nesta longa jornada.

A Gílton José Ferreira da Silva, pelo jeito descontraído e competente de nos orientar neste trabalho e a todos os professores que contribuíram para chegarmos até aqui. Professores vocês ensinam mais que uma profissão, ensinam para a vida, muitíssimo obrigada.

¹ Mais informações em <http://mundoestranho.abril.com.br/cultura/quantos-idiomas-existem-no-mundo/>

*Até prova em contrário, todas as coisas são possíveis - e mesmo
o impossível talvez o seja apenas nesse momento.*
(Pearl S. Buck)

Resumo

O mundo está vivenciando duas grandes transformações: o número de pessoas vivendo em zonas urbanas ultrapassando o número de pessoas em zonas rurais e uma revolução tecnológica mudando a vida de bilhões de pessoas. Com mais pessoas vivendo nas cidades, os problemas que elas enfrentam são intensificados e a resposta a esses problemas envolve cada vez mais o emprego de muita tecnologia, de onde emerge o conceito de Cidade Inteligente. Com o intuito de lidar com um dos problemas mais cruciais das cidades, a mobilidade urbana, este trabalho procurou levantar as suas causas e quais intervenções poderiam ser realizadas. Um dos problemas mais proeminentes que gera impacto na mobilidade urbana é o crescimento da quantidade de veículos circulando nas cidades e um dos desafios é gerenciar esta frota de veículos buscando diminuir o congestionamento, facilitando a mobilidade e consequentemente reduzindo a quantidade de gases expelidos na atmosfera. Uma vez que uma das causas deste problema é o tempo gasto pelos motoristas na procura por vagas para estacionar, uma solução é minimizar esta causa. Para este fim foi realizada uma Revisão Sistemática dos estudos primários que estão inseridos neste contexto e por meio de uma Pesquisa de Produtos no Mercado, seguido da aplicação de um questionário para *stakeholders* foram extraídas características relevantes para o desenvolvimento de uma solução para este cenário. Sendo assim o objetivo deste trabalho foi criar uma ferramenta que otimiza de forma inteligente o processo de oferta e procura de vagas de estacionamento, tendo em vista o número crescente de veículos e o espaço limitado das cidades como contribuição para o desenvolvimento das Cidades Inteligentes.

Palavras-chave: estacionamento inteligente, aplicativos móveis, cidades inteligentes, mobilidade urbana.

Abstract

The world is experiencing two major transformations: the number of people living in urban areas surpassing the number of people in rural areas and a technological revolution changing the lives of billions of people. With more people living in cities, the problems they face are intensified and the response to these problems increasingly involves the use of a lot of technology, from which emerges the concept of Smart City. In order to deal with one of the most critical problems of cities, urban mobility, this work seeks to understand its causes and what interventions could be undertaken. One of the most prominent problems that impacts the urban mobility is the high number of vehicles circulating in cities and one of the challenges is to manage this fleet of vehicles in order to reduce congestion, facilitating mobility and consequently reducing the amount of exhaust gases in the atmosphere. Since one of the causes of this problem is the time spent by drivers looking for parking spaces, one solution is to minimize it. For this purpose a Systematic Literature Review of primary studies that were inserted in this context was carried out and through a Product Research in the Market, followed by the application of a questionnaire to stakeholders, relevant characteristics were extracted for the development of a solution for this scenario. Therefore, the objective of this work was to create a tool that intelligently optimizes the process of supply and demand of parking spots, considering the growing number of vehicles and the limited space of cities as a contribution to the development of Smart Cities.

Keywords: *smart parking, mobile application, smart cities, urban mobility*

Lista de ilustrações

Figura 1 – Modelo em Cascata	24
Figura 2 – Fluxo de processo Scrum	25
Figura 3 – Quantidade de Artigos coletados por base	33
Figura 4 – Artigos selecionados, rejeitados e duplicados durante seleção individual - Autor 1 e Autor 2.	34
Figura 5 – Artigos aceitos e rejeitados na fase de seleção conjunta	35
Figura 6 – Quantidade de artigos por critério de inclusão	36
Figura 7 – Telas do aplicativo Parking: Localizar meu Carro	46
Figura 8 – Telas do aplicativo ParAli Estacionamento	47
Figura 9 – Telas do aplicativo ParkApp Admin Estac. e Valets	48
Figura 10 – Telas do aplicativo Himachal Parking	50
Figura 11 – Telas do aplicativo Onde Parar	51
Figura 12 – Telas do aplicativo TelPark	52
Figura 13 – Telas do aplicativo Parking Aki	53
Figura 14 – Telas do aplicativo ParkMe	54
Figura 15 – Gráfico das respostas da pergunta 1 do questionário para levantamento de requisitos do Smart Parking.	57
Figura 16 – Gráfico das respostas da pergunta 4 do questionário para levantamento de requisitos do Smart Parking.	58
Figura 17 – Gráfico das respostas da pergunta 7 relativo aos aplicativos utilizados na procura por vagas de estacionamento do questionário para levantamento de requisitos do Smart Parking.	59
Figura 18 – Gráfico das respostas da pergunta 9 do questionário para levantamento de requisitos do Smart Parking.	60
Figura 19 – Ambiente de desenvolvimento - Android Studio	62
Figura 20 – Diagrama de Casos de Uso - Smart Parking	66
Figura 21 – Principais telas do protótipo de telas do Smart Parking	67
Figura 22 – Diagrama de Hierarquia de Telas - Smart Parking	68
Figura 23 – Questionário de validação do protótipo de Telas - Tela de Cadastro	69
Figura 24 – Hierarquia de Telas do protótipo desenvolvido	70
Figura 25 – Organização da Arquitetura seguindo as diretrizes MVP	72
Figura 26 – Diagrama de Pacotes - Arquitetura do aplicativo proposto baseado no padrão MVP	73
Figura 27 – Diagrama de Classes - SmartParking (Parte 1)	74

Figura 28 – Diagrama de Classes - SmartParking (Parte 2)	75
Figura 29 – Diagrama de Classes - SmartParking (Parte 3)	76
Figura 30 – Diagrama de Classes - SmartParking (Parte 4)	77
Figura 31 – Modelo Json - Firebase Realtime Database	78
Figura 32 – Tela do Trello - Planejamento das Sprints	80
Figura 33 – Árvore de diretórios do projeto	82

Lista de quadros

1	Critérios para definição da questão de pesquisa	30
2	Termos e Sinônimos em Inglês	31
3	String generica de busca	32
4	<i>String</i> de busca de cada base	33
5	Identificação dos Estudos	36
6	Síntese dos pontos mais relevantes extraídos nos estudos	38
7	Identificação das características	44
8	Aplicativos e URLs para acesso	45
9	String de busca utilizada na pesquisa do Parking: Localizar meu carro	46
10	String de busca utilizada na pesquisa do ParAli Estacionamento	47
11	String de busca utilizada na pesquisa do ParkApp Admin Estac. e Valets	49
12	String de busca utilizada na pesquisa do Himachal Parking	50
13	String de busca utilizada na pesquisa do OndeParar	51
14	<i>String</i> de busca utilizada na pesquisa do Telpark	52
15	<i>String</i> de busca utilizada na pesquisa do ParkingAki	54
16	String de busca utilizada na pesquisa do ParkMe	55
17	Principais características identificadas nos aplicativos	55
18	Relação entre os aplicativos avaliados e as características associadas	56
19	Configurações do notebook Lenovo	63
20	Configurações do notebook Dell	63
21	Requisitos Funcionais	64
22	Requisitos Não-Funcionais	65
23	Histórias Implementadas	81

Lista de códigos

Código 1 – Contract - Interface <code>VehicleManagerContract</code>	82
Código 2 – View - <code>VehicleManagerActivity</code>	83
Código 3 – Presenter - <code>VehicleManagerPresenter</code>	84

Lista de abreviaturas e siglas

ABC & S	Always Best Connected and Best Served
API	Application Programming Interface
APK	Android Package
AU	Agente de Usuário
CEP	Código de Endereçamento Postal
CSS	Cascading Style Sheets
EC	Exclusion Criteria
EUA	Estados Unidos da América
GE	Gestor de Estacionamento
HGASP	Hybrid Genetic Assignment Search Procedure
HTC	High Tech Computer Corporation
HTTP	HyperText Transfer Protocol
IBM	International Business Machines
IC	Inclusion Criteria
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronic Engineers
IOS	Iphone Operating System
IOT	Internet Of Things
JDK	Java Development Kit
JSON	JavaScript Object Notation
M2M	Máquina Para Máquina
NFC	Near Field Communication
ONU	Organização das Nações Unidas
OSGi	Open Services Gateway initiative

SMS	Short Message Service
SO	Sistema Operacional
SRM	Sistema de Recomendação de Mobilidade
GPS	Global Positioning System
GRASP	Greedy Randomized Adaptive Search Procedure
MVC	Model-View-Controller
RS	Revisão Sistemática
SQL	Structured Query Language
SSGA	Single Seed-Growing Algorithm
TIC	Tecnologia da Informação e Comunicação
UFS	Universidade Federal de Sergipe
UI	User Interface
UML	Unified Modeling Language
WSN	Wireless Sensor Network

Sumário

1	Introdução	18
1.1	Objetivos	19
1.1.1	Geral	19
1.1.2	Específicos	19
1.2	Metodologia	20
1.3	Estrutura do Documento	20
2	Fundamentação Teórica	21
2.1	Cidades Inteligentes	21
2.2	Mobilidade Urbana	22
2.3	Desenvolvimento de Software	22
2.3.1	Abordagem Tradicional	23
2.3.2	Abordagem Ágil	24
2.3.2.1	Scrum	25
2.4	Aplicações Móveis	26
2.4.1	Android	27
3	Trabalhos Relacionados	29
3.1	Revisão Sistemática dos estudos primários	29
3.1.1	Planejamento da revisão	30
3.1.2	Questões de Pesquisa	30
3.1.3	Critérios de Inclusão e Exclusão	30
3.1.4	String de Busca	31
3.1.5	Seleção dos estudos primários	32
3.1.6	Extração	35
3.2	Revisão de Produtos no Mercado	45
3.2.1	Parking: Localizar meu carro	45
3.2.2	ParAli Estacionamento	47
3.2.3	ParkApp Admin Estac. e Valets	48
3.2.4	Himachal Parking	49
3.2.5	Onde Parar	50
3.2.6	Telpark	52
3.2.7	ParkingAki	53
3.2.8	ParkMe	54
3.3	Questionário com possíveis utilizadores para levantamento de requisitos	56

4	Desenvolvimento do Produto de Software	61
4.1	Ferramentas Utilizadas	61
4.2	Configuração da máquina de desenvolvimento	63
4.3	Requisitos de Software	63
4.3.1	Requisitos Funcionais	64
4.3.2	Requisitos Não-Funcionais	65
4.4	Diagrama de Casos de Uso	65
4.5	Protótipo de Telas	66
4.5.1	Hierarquia de Telas do projeto	68
4.5.2	Validação do protótipo de telas	69
4.5.3	Hierarquia de Telas do Produto de Software	70
4.6	Arquitetura	71
4.7	Diagrama de Classes	73
4.8	Esquema do Banco de Dados	77
4.9	Planejamento das Sprints	78
4.10	Implementação do Protótipo do Produto	80
4.11	Validação do Aplicativo	85
4.11.1	Questionários de Validação	85
4.11.2	Testes Realizados	86
5	Considerações Finais e	
	Trabalhos Futuros	87
	Referências	90
	Apêndices	96
APÊNDICE A	Questionário com <i>stakeholders</i>	97
APÊNDICE B	Questionário para validação do Protótipo de Telas do Gestor de Estacionamento	101
APÊNDICE C	Questionário para validação dos Protótipo de Telas do Motorista	111
APÊNDICE D	Questionário para validação do aplicativo desenvolvido - Usuário Gestor de Estacionamento	122
APÊNDICE E	Questionário para validação do aplicativo desenvolvido - Usuário Motorista	127

APÊNDICE F	Projeto de Testes Smart Parking	133
APÊNDICE G	Manual do Usuário do Smart Parking	169

1

Introdução

Em 1950, 746 milhões de pessoas viviam em áreas urbanas, em 2014, esse número já chegava a 3.9 bilhões, correspondendo a 54 % da população mundial, estima-se que em 2050, 66% das pessoas vivam em áreas urbanas (UN, 2014b). Essa rápida urbanização trouxe consigo vários desafios que costumam se concentrar em cinco grandes áreas: meio ambiente, mobilidade, interação cidadão-governo, qualidade de vida e economia (PRESS; BRAZIL, 2017).

Para lidar com desafios como estes, as cidades precisam fazer mais do que absorver alguma dose de tecnologia, elas precisam se tornar "inteligentes". Uma Cidade Inteligente (smart city) é aquela que supera os desafios do passado e conquista o futuro, utilizando a tecnologia como um meio para prestar de forma mais eficiente os serviços urbanos e melhorar a qualidade de vida dos cidadãos (CUNHA et al., 2016). Isso significa, utilizar tecnologia e criatividade para resolver seus problemas.

Um desses desafios, a mobilidade urbana, tem ganhado cada vez mais proeminência. Em 2008, o mundo atingiu o impressionante número de 1 bilhão de veículos circulando (DAVIS; WILLIAMS; BOUNDY, 2016). No Brasil, em 2015, eram 81,6 milhões de automóveis em circulação (REPOSITORY, 2015), quase um para cada dois habitantes. Lidar com esse número grande e crescente de veículos, especialmente em horários de pico, deixou de ser um problema apenas das grandes cidades para ser uma realidade de cada aglomeração urbana.

E um dos aspectos mais críticos nessa questão é a falta do espaço: como acomodar uma frota crescente de veículos nos espaços urbanos cada vez mais disputados? A solução passa por gerenciar com inteligência o fluxo de veículos, tendo a tecnologia como aliada na busca de mais eficiência em mobilidade urbana.

Assim, uma das maneiras inteligentes de se lidar com essa questão é evitar que os motoristas fiquem vagando pela cidade desnecessariamente, por exemplo, quando estão à procura de disputadas vagas de estacionamento. E se o motorista pudesse apenas informar o destino

e receber de volta as informações sobre quais estacionamentos na região de destino possuem vagas, de acordo com seus critérios? Ele pouparia tempo, otimizaria a utilização dos espaços para veículos na cidade, ajudaria a reduzir o tráfego e até economizaria combustível, reduzindo também a emissão de poluentes.

Com a disseminação e o crescimento das tecnologias móveis e pervasivas, a otimização da busca por um bom local para estacionar torna-se possível, abrindo espaço para o desenvolvimento de aplicações que a auxiliem.

Este trabalho visa desenvolver uma aplicação que promove a ocupação de vagas de estacionamento de forma inteligente, facilitando o processo de busca destes espaços e atendendo tanto as necessidades dos motoristas quanto as dos donos de locais de estacionamento.

1.1 Objetivos

A partir do conhecimento acerca da problemática vivenciada pelos motoristas que buscam vagas para estacionar e com o intuito de contribuir para o desenvolvimento das cidades inteligentes, os objetivos geral e específico deste trabalho são:

1.1.1 Geral

Criar uma ferramenta para otimizar de forma inteligente o processo de oferta e procura de vagas de estacionamento tendo em vista o número crescente de veículos e o espaço limitado das cidades.

1.1.2 Específicos

A fim de alcançar o objetivo acima citado, este trabalho tem como objetivos específicos:

- Realizar um levantamento das necessidades do cenário estudado por meio de uma Revisão Sistemática dos trabalhos registrados em bases acadêmicas que tratem do gerenciamento de vagas para estacionar, de uma Pesquisa de Produtos de Mercado e da aplicação de um questionário com possíveis utilizadores da ferramenta;
- Desenvolver um protótipo funcional afim de validar as características e a partir deste desenvolver um aplicativo para auxiliar os condutores a encontrar vagas de estacionamento de acordo com os seus critérios bem como, ajudar os proprietários de estacionamentos a gerenciarem seu espaço de maneira inteligente;
- Proteger a Propriedade Intelectual gerada neste trabalho.

1.2 Metodologia

Com base nos objetivos estabelecidos esta pesquisa pode ser classificada como pesquisa exploratória, um tipo de pesquisa que segundo Gil (2007, p.41), "tem o objetivo de proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a construir hipóteses".

A metodologia aplicada no desenvolvimento deste trabalho tem natureza de pesquisa aplicada, a qual segundo Engel e Silveira (2009, p.35) "tem como objetivo gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos. Envolve verdades e interesses locais". Também se caracteriza como pesquisa bibliográfica, devido ao levantamento bibliográfico realizado na Revisão Sistemática proposta, tendo como base material já elaborado, constituído principalmente de livros e artigos científicos (GIL, 2007). Assim como também pode ser classificada como *survey*, tendo em vista a utilização de questionários, os quais para Fonseca (2002, p.33) visam "a obtenção de dados ou informações sobre as características ou as opiniões de determinado grupo de pessoas, indicado como representante de uma população-alvo, utilizando um questionário como instrumento de pesquisa".

1.3 Estrutura do Documento

Para facilitar a navegação e melhor entendimento, este documento está estruturado em capítulos e seções, que são:

- Capítulo 1 - Introdução: Esta seção contextualiza acerca da problemática abordada, assim como os objetivos e a metodologia utilizada;
- Capítulo 2 - Fundamentação Teórica: Esta seção define conceitos relevantes a assimilação do trabalho proposto;
- Capítulo 3 - Trabalhos Relacionados: Esta seção descreve o levantamento de informações obtidas por meio da Revisão Sistemática e Revisão de Mercado realizadas e da aplicação do questionário;
- Capítulo 4 - Desenvolvimento: Esta seção apresenta o projeto da aplicação proposta, descrevendo as ferramentas utilizadas, os Requisitos Funcionais e Não Funcionais da aplicação, o diagrama de Casos de Uso, além do Protótipo e a Hierarquia de Telas, o diagrama de Classes, a Arquitetura utilizada, o esquema do Banco de Dados, o planejamento das Sprints e a validação do aplicativo;
- Capítulo 5 - Considerações Finais e Trabalhos Futuros: Esta seção apresenta os objetivos alcançados, a relevância do trabalho proposto e a expectativa para os trabalhos futuros.

2

Fundamentação Teórica

Esse capítulo apresenta os conceitos relevantes na compreensão do trabalho realizado, tendo como objetivo situar o leitor no contexto a que este trabalho se refere.

2.1 Cidades Inteligentes

Em 1950, apenas 30% da população mundial vivia em áreas urbanas, já em 2014, 54% da população mundial residia nessas áreas. E apesar do número de megalópolis (cidades com mais de 10 milhões de habitantes) ter subido para 28 ao redor do mundo em 2014, elas concentram apenas um de cada oito habitantes de áreas urbanas. Quase metade dos moradores das zonas urbanas vive em cidades relativamente pequenas (com menos de 500 mil habitantes) e que costumam apresentar crescimento populacional acelerado (UN, 2014a).

Isso indica que o mundo está se tornando cada vez mais urbano, e essa tendência deve se manter, em virtude da migração de áreas rurais para áreas urbanas que ocorre em países em desenvolvimento. Em razão disso, "Gerenciar áreas urbanas se tornou um dos mais importantes desafios para o desenvolvimento no século XXI." Afirma John Wilmoth, Diretor da Divisão de população do Departamento de assuntos econômicos e sociais da Organização das Nações Unidas(ONU) (UN, 2014b, p.5).

Contudo, a urbanização não é única tendência transformando a sociedade contemporânea, paralelamente, vivemos uma Revolução Digital. Tal como nunca antes, o desenvolvimento das Tecnologias da Informação e Comunicação (TIC), fez com que se disseminassem os dispositivos fixos e móveis conectados, a chamada hiperconectividade, tanto entre pessoas como entre máquinas (M2M), os quais transformaram o modo como se articulam o sistema produtivo e a sociedade. O impacto da revolução digital sobre um mundo em acelerada urbanização sustenta a emergência do conceito *smart city* (CUNHA et al., 2016).

Para [Caragliu, Bo e Nijkamp \(2009\)](#) uma cidade é Inteligente quando investe-se em capital humano e social e em uma infraestrutura de comunicação tradicional e moderna (TIC) que impulsiona um crescimento econômico sustentável e uma alta qualidade de vida, e possui também uma gestão sábia dos recursos naturais, por meio de uma governança participativa.

Uma *smart city* (cidade inteligente) é uma cidade em que a sociedade e o governo utilizam juntos a tecnologia para promover desenvolvimento sustentável e melhorar a qualidade de vida, impactando positivamente a economia, a mobilidade urbana, o meio ambiente, a educação, a saúde, a segurança pública e a gestão da cidade ([IEEE, 2014](#)).

2.2 Mobilidade Urbana

A Mobilidade Urbana pode ser definida como a totalidade das viagens realizadas diariamente pelos habitantes de uma cidade e os métodos e condições associadas a tais viagens (meios de transporte selecionados, duração das viagens, tempo gasto no transporte, etc.) ([SHANG; DOULET; KEANE, 2009](#)).

Com mais pessoas vivendo em cidades, aumenta o número de deslocamentos feitos dentro delas. Estima-se que 64% de todos os quilômetros percorridos em viagens em 2014 foram em deslocamentos internos. Uma consequência inevitável do crescimento populacional acelerado combinado com um sistema de mobilidade urbana sem reformas e com pouco investimento é o engarrafamento. Em 2014, o tempo médio que um morador de uma área urbana passou em engarrafamentos foi de 53 horas por ano. Em 2050, mantendo-se as tendências atuais, o tempo médio perdido em engarrafamentos será de 106 horas por ano. Isso aliado a crescente poluição gerada nas cidades, da qual, os meios de transporte tem uma contribuição significativa, impacta negativamente a qualidade de vida do cidadão comum ([AUDENHOVE et al., 2014](#)).

Um dos fatores mais importantes na Mobilidade Urbana é a Gestão dos Locais de Estacionamento, pois isso influencia diretamente a segurança rodoviária, a eficiência operacional e a ordem no trânsito. De acordo com [Rossetti \(2016\)](#):

Entender as complexidades e importância do planejamento para estacionamento de veículos é vital no contexto das cidades inteligentes. Na verdade, de acordo com Todd Litman, "os detalhes de regulamentos de estacionamento podem realmente ter amplos impactos sobre a vida da cidade, reduzindo o tráfego e a poluição e até aumentando as receitas locais". Sistemas para gestão de estacionamento podem se beneficiar grandemente dos avanços tecnológicos proporcionados pela IOT, promovendo a inovação por meio de serviços que fazem uso intensivo de dados, de modo a melhorar a mobilidade urbana sustentável.

2.3 Desenvolvimento de Software

Desde o seu surgimento até os dias atuais, a Computação, vem sendo incorporada ao dia-a-dia e tem se tornado ubíqua. É cada vez mais comum o uso de dispositivos de computação

conectados à internet interagindo com o mundo físico e se tornando indispensável.

Para que isso seja possível, existe um componente que é essencial para que todo sistema computacional funcione, o software, o qual segundo [Pressman \(2011, p.32\)](#):

Consiste em um conjunto composto por instruções (programas de computador) que fornecem características de funções e desempenho desejado, estruturas de dados que possibilitam aos programas manipular informações adequadamente e documentos, os quais descrevem a operação e o uso do programa.

Como área de conhecimento, o software possui uma engenharia, a Engenharia de Software, que segundo a [IEEE \(2010, p.331\)](#) é a aplicação de abordagens sistemáticas, disciplinadas e quantificáveis ao desenvolvimento, operação e manutenção de software, além do estudo dessas abordagens. A Engenharia de Software define modelos de processo para o desenvolvimento de software e suas tarefas principais que consistem em, especificação, desenvolvimento, validação e evolução do software.

Na especificação do software, os clientes e engenheiros definem o software a ser produzido e suas restrições. Em seu desenvolvimento, o software é projetado e programado. Na validação, o software é verificado para garantir que atenda as necessidades do cliente e por fim, a evolução do software, é uma atividade na qual o software é modificado para se adaptar às mudanças do requisito do cliente e do mercado ([SOMMERVILLE, 2007](#)).

Os processos de desenvolvimento de software podem ser classificados em duas abordagens: A Tradicional e a Ágil

Na abordagem Tradicional, o desenvolvedor deve ser um indivíduo hábil, o cliente é pouco envolvido com o projeto e os requisitos são conhecidos e estáveis. Caso seja necessário mudanças nos requisitos existe um retrabalho custoso para que a mudança seja aplicada. Ainda, nesta abordagem o planejamento redireciona os resultados, o foco é em grandes projetos e o objetivo é controlar, em busca de alcançar o planejado.

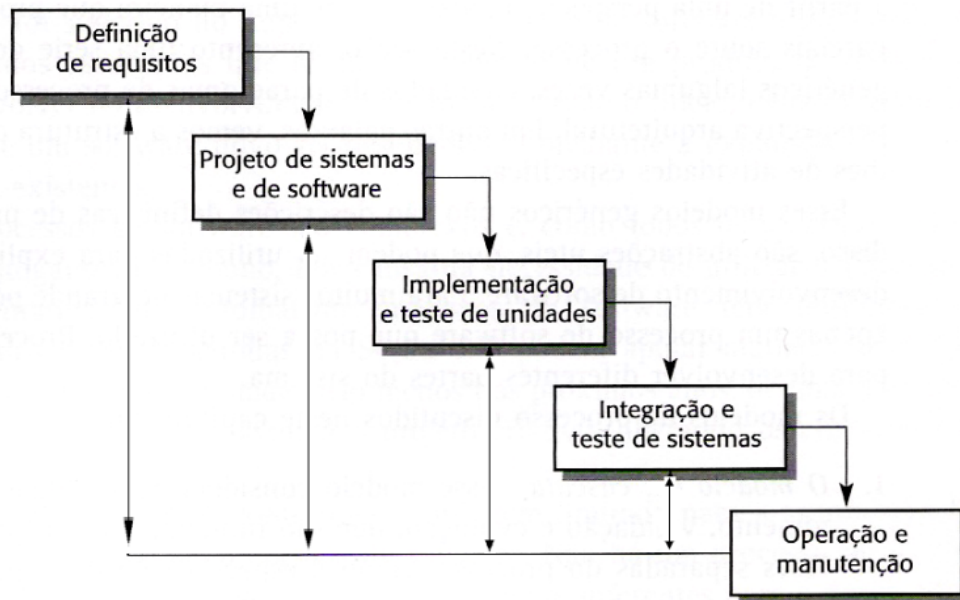
Na abordagem Ágil, por outro lado, o desenvolvedor é um indivíduo ágil, o cliente deve estar comprometido com o projeto acompanhando e participando das decisões, os requisitos são emergentes e mutáveis, devido a dinamicidade dos cenários nos quais esta abordagem é aplicada e em caso de mudanças de requisitos, o retrabalho é mais barato. Os resultados redirecionam o planejamento, o foco é em projetos de natureza exploratória e inovadora e o objetivo é simplificar o processo de desenvolvimento.

2.3.1 Abordagem Tradicional

Esta abordagem é baseada em um modelo de processo em cascata, o qual considera as atividades principais do processo - especificação, desenvolvimento, validação e evolução do software - como sendo fases separadas e sequenciais. Neste modelo, as 4 atividades principais são divididas em 5 fases - definição de requisitos, projeto de sistemas e de software, implementação e

teste de unidades, integração e teste de sistemas, operação e manutenção - como está demonstrado na Figura 1.

Figura 1 – Modelo em Cascata



Fonte: (SOMMERVILLE, 2007)

O modelo em cascata exige que todos os requisitos sejam definidos completamente na fase inicial do processo e que o cliente esteja comprometido com estes requisitos antes do projeto começar, assim como o projetista deve estar comprometido com estratégias específicas de projeto antes da implementação. Neste cenário, mudanças nos requisitos requer retrabalho dos requisitos e do projeto. Por outro lado produz um sistema com uma maior documentação (SOMMERVILLE, 2007).

2.3.2 Abordagem Ágil

Surgiu como uma adaptação do desenvolvimento Tradicional apto a lidar com as mudanças frequentes nas exigências dos mercados, cada vez mais dinâmicos, fruto do "Manifesto para Desenvolvimento Ágil de Software" criado por Kent Beck e outros dezesseis estudiosos da área de Engenharia de Software em 2001. O desenvolvimento ágil:

defende a satisfação do cliente e a entrega incremental prévia; equipes de projeto pequenas e altamente motivadas; métodos informais; artefato de engenharia de software mínimos e, acima de tudo, simplicidade no desenvolvimento geral. Os princípios de desenvolvimento priorizam a entrega mais que análise e projeto (embora estas atividades não sejam desencorajadas); também priorizam a comunicação ativa e contínua entre desenvolvedores e clientes" (PRESSMAN, 2011, p.81).

As metodologias tradicionais não focam em um fator determinante, o fator humano. O desenvolvimento de software ágil, por outro lado, trata as interações humanas como parte

central para o sucesso do projeto e a incorporação da efetiva agilidade necessária no processo de desenvolvimento.

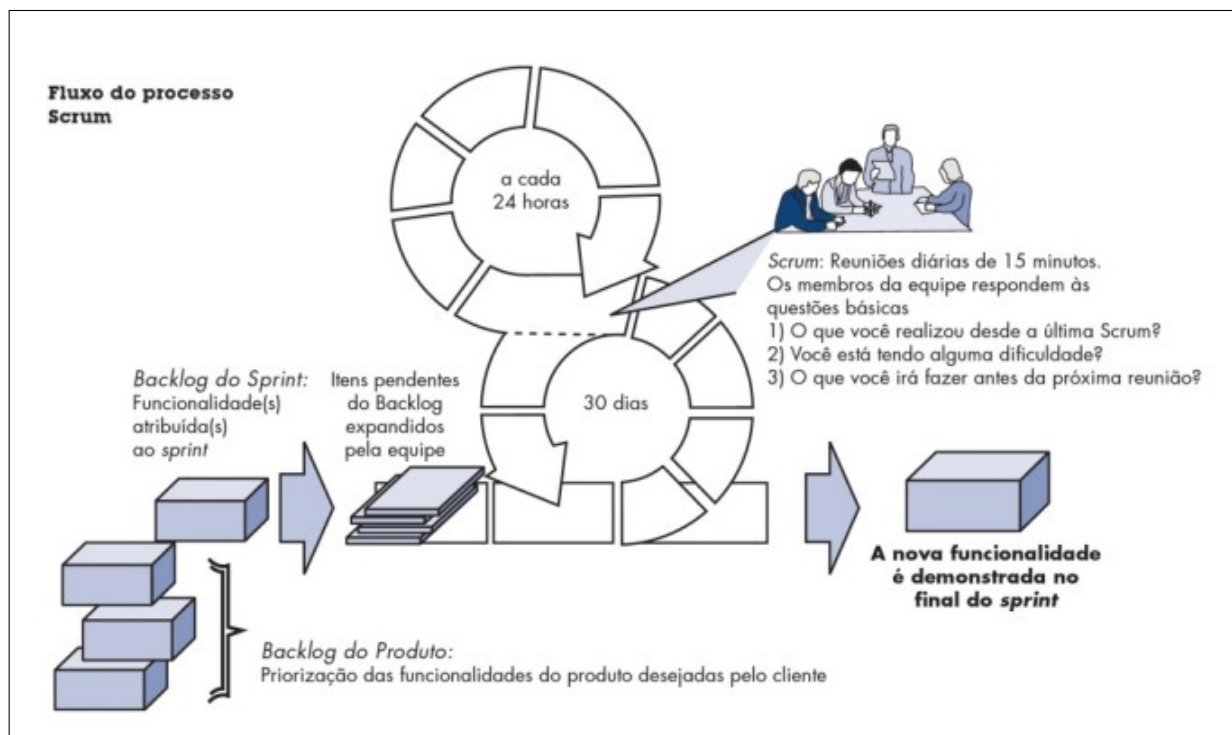
Existem diferentes abordagens de desenvolvimento de software ágil entre elas existe o Scrum, metodologia apresentada na seção seguinte.

2.3.2.1 Scrum

O "Scrum é um *framework* dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível"(SCHWABER; SUTHERLAND, 2013, p.3).

O Scrum possui três papéis principais: O *Product Owner*, o *Scrum Master* e o Time de Desenvolvimento. O *Product Owner* tem a responsabilidade de cuidar das necessidades do produto, é o elo do cliente com o time e encarregado de apresentar os interesses de todos os *stakeholders*. O *Scrum Master*, cuida do processo e verifica se tudo dentro do projeto está ocorrendo de maneira correta, funciona como um treinador. Treina, mas não tem autoridade sobre a equipe, sabe tudo sobre o Scrum e auxilia as pessoas que não tem tanta experiência. O Time de Desenvolvimento, por sua vez, tem que implementar o produto, podendo ser uma equipe formada por entre 3 a 9 pessoas, e deve ser auto-gerenciado. O fluxo do processo utilizado na metodologia Scrum acontece como apresentado na Figura 2.

Figura 2 – Fluxo de processo Scrum



Fonte: (PRESSMAN, 2011)

O Scrum (Figura 2) funciona de forma que inicialmente é definido o *backlog* do produto (*product backlog*), uma lista priorizada contendo as principais características que o produto deve conter, ou seja, o que mais interessa para o cliente no produto. É um artefato que pode mudar dependendo da necessidade do cliente.

A partir do *backlog* do produto, acontece a reunião de planejamento da *sprint*, na qual é determinado todo o trabalho que será executado naquela *sprint*, ou seja, o que vai ser entregue no final dela, nesta reunião são escolhidos alguns itens do backlog do produto para serem trabalhados durante a *sprint*. Uma *sprint* é um período de tempo (usualmente entre 1 a 4 semanas), no qual a quantidade de trabalho é executada. Cada *sprint* cria um artefato, o Backlog da Sprint (*Sprint Backlog*), o qual contém apenas os itens daquela *sprint* que está se iniciando.

A próxima fase do Scrum é o trabalho de desenvolvimento do produto, no qual todos os dias durante a *sprint*, além da implementação do artefato, são realizadas reuniões, as quais servem para fazer uma inspeção do trabalho realizado dentro da equipe de desenvolvimento, pergunta-se o que foi realizado desde a ultima Scrum, se está tendo alguma dificuldade e o que será feito antes da próxima reunião. Não deve durar mais que 15 minutos.

Ao final de cada *sprint* tem um incremento, este incremento vai para a *sprint review*, uma reunião de revisão ou reunião de demonstração. Nesta reunião o *Product Owner* define se aceita ou não aquele produto. Outro ponto importante nesta reunião é recolher *feedback* sobre o produto, o qual é usado para refinar este ultimo. Esta reunião deve durar no máximo 4 horas.

Por fim, a reunião de retrospectiva, ocorre depois da reunião de revisão. Nesta a equipe vai refletir o que foi que aconteceu de bom e o que aconteceu de ruim criando um plano de melhoria para ser aplicado nas próximas *sprints*, isso garante que a equipe tenha uma melhoria e um refinamento contínuo nos seus processos (BERNARDO, 2015).

2.4 Aplicações Móveis

As aplicações móveis são aquelas desenvolvidas para serem executadas em dispositivos móveis como *smartphones* e tablets. Esses dispositivos tem se tornado ferramentas indispensáveis ao cotidiano de um número cada vez mais maior de pessoas. Com mais pessoas os utilizando tem crescido o número de aplicativos existentes nas plataformas onde eles são disponibilizados. Uma pesquisa realizada pela Statista (2017) mostrou que o número de aplicativos disponíveis em março de 2017 atingiu os 2,8 milhões, na *Play Store* da Google e 2,2 milhões na *App Store* da Apple. Juntas, Apple e Google, respondem quase pela totalidade do mercado de dispositivos móveis.

Embora o desenvolvimento de aplicações para dispositivos móveis remonte os anos 90, de acordo com Wasserman (2010) o crescimento destes números subiu exponencialmente

em 2008, ano em que a Apple¹ lançou a App Store², loja de aplicativos para o iPhone, e a Google³ lançou o Android, que também seria posteriormente acompanhado de sua própria loja de aplicativos, a Google Play⁴.

Embora, os aplicativos voltados para dispositivos móveis tenham em seu desenvolvimento certa semelhança com outras aplicações embarcadas, eles apresentam desafios adicionais, como por exemplo, o uso de sensores e a necessidade de lidar com uma quantidade limitada de recursos (memória, processamento, espaço em disco, etc.) se comparada com aplicações desktop.

Existe também uma preferência no uso de métodos ágeis para desenvolvimento de aplicativos, Wasserman (2010) sugere, baseado em uma pesquisa com desenvolvedores de aplicativos, que até desenvolvedores individuais tem utilizado o Scrum como processo no desenvolvimento de aplicações mobile.

2.4.1 Android

Sistema Operacional Móvel desenvolvido pela Google, baseado no *kernel* Linux e projetado principalmente para dispositivos móveis sensíveis ao toque, como smartphones e tablets. A interface do usuário do Android é baseada, principalmente, na manipulação direta, fazendo uso de gestos que correspondem vagamente a ações do mundo real para manipular objetos na tela e também de um teclado virtual para entrada de texto. Além de celulares e tablets, o Google desenvolveu ainda o Android TV para televisores, o Android Auto para carros e o Android Wear para "dispositivos vestíveis"(como *smartwatches*), cada um com uma interface de usuário específica.

Em 2003, na cidade de Palo Alto, na Califórnia foi fundada por Andy Rubin, Rich Miner, Nick Sears e Chris White a *Android Inc* (PRADO, 2017). A companhia se concentrou na evolução de código aberto⁵ de muitas das ideias que começaram na *Danger* (antiga empresa de Rubin), almejando oferecer a melhor experiência conectada à Web possível para usuários mobile e criando um ambiente em que qualquer desenvolvedor pudesse colaborar (DOBIE et al.,).

Em 2005, a Android Inc. foi comprada pela Google, que preparava-se para entrar no mercado de dispositivo móveis, já que Larry Page e Sergey Brin (fundadores e donos da Google) desejavam que mais celulares tivessem o Google como o mecanismo de busca padrão (MANJOO, 2015). À época o mercado era dominado pela Blackberry⁶ e pela Microsoft⁷. Em 2007, a Apple causou um grande impacto com o lançamento do iPhone. Isso, acabou por adiar o lançamento do

¹ Apple: <https://www.apple.com/>

² App Store da Apple: <https://www.appstore.com/>

³ Google: <https://www.google.com/intl/pt-BR/about/>

⁴ Google Play: play.google.com/Store

⁵ Código aberto é um modelo de desenvolvimento que promove um licenciamento livre para o design ou esquematização de um produto, e a redistribuição universal desse design ou esquema, dando a possibilidade para que qualquer um consulte, examine ou modifique o produto(JUNIOR, 2003).

⁶ Blackberry: <https://us.blackberry.com/>

⁷ Microsoft: <https://www.microsoft.com/pt-br/>

Android. O primeiro smartphone a chegar ao mercado com o Android foi o HTC Dream, lançado em setembro de 2008 (DOBIE et al.,).

Em 2016, as vendas de *smartphones* com o sistema operacional Android corresponderam a 85,2% do total de *smartphones* vendidos (DUNN, 2016). Em 2017, 65,19% de todos os dispositivos móveis rodavam o Sistema Operacional Android, mais que o dobro do que detinha o segundo colocado, o iOS da Apple (SHARE, 2017). Com mais de um bilhão de dispositivos rodando o Android sendo vendidos anualmente desde 2014, ele tornou-se não apenas o Sistema Operacional mais popular apenas entre os dispositivos móveis, mas o S.O. mais popular no geral (MANJOO, 2015).

3

Trabalhos Relacionados

Este trabalho explorou três diferentes pontos de vista. Inicialmente foi realizada uma investigação do estado da arte por meio de uma pesquisa em bases de trabalhos científicos, posteriormente foram exploradas quais as soluções que já existem no mercado e por fim, foi conhecida a perspectiva do usuário, por meio da aplicação de um questionário que teve como objetivo fazer o levantamento das necessidades destas pessoas para a partir deste levantamento elaborar uma solução que busca sanar estas necessidades. Esse capítulo apresenta a Revisão Sistemática realizada, assim como a Revisão dos Produtos de Mercado e o questionário para levantamento de requisitos, resultado do trabalho de pesquisa realizado tendo como objetivo extrair informações relevantes para o planejamento do produto de software proposto.

3.1 Revisão Sistemática dos estudos primários

De forma geral Revisão Sistemática (RS) na área científica se refere ao ato de rever todo trabalho científico relativo a um fenômeno de interesse de forma ordenada. Essa ordenação implica o uso de uma estratégia para procura e seleção dos trabalhos desejados. [Kitchenham \(2004, p.1\)](#) define a RS como um meio de identificar, avaliar e interpretar toda a pesquisa disponível relevante para uma questão de pesquisa em particular, tópico ou fenômeno de interesse e propõe uma abordagem definida em três fases principais, sendo estas, o planejamento da revisão, a condução da revisão e a apresentação da revisão.

O planejamento da revisão trata da identificação do que é necessário para a revisão e o desenvolvimento de um protocolo que a direcione. A condução da revisão é a identificação da pesquisa, seleção dos estudos primários, avaliação da qualidade dos estudos, extração e síntese dos dados e por fim a apresentação da revisão é a documentação de todo o trabalho realizado. Trabalhos assim sumariza o que há de uma determinada área e são ferramentas que muito agrega a trabalhos acadêmicos em geral.

O objetivo desta revisão é pesquisar trabalhos já existentes relacionados a aplicações de estacionamento para cidades inteligente, na tentativa de mapear o que já está consolidado na área e selecionar aqueles que sejam adequados ao trabalho proposto.

3.1.1 Planejamento da revisão

Na fase de planejamento, baseando-se em [Kitchenham \(2004\)](#), foi definido o protocolo usado na condução da Revisão Sistemática partindo da elaboração da pergunta de pesquisa para a fase da definição de critérios de inclusão e exclusão e por fim para a definição da *string* de busca genérica.

3.1.2 Questões de Pesquisa

Para a construção das questões de pesquisa foram considerados os três pontos de vista sugeridos por [Kitchenham \(2004\)](#): População(Population), Intervenção(Intervention) e Resultados(Outcomes), os quais estão apresentados no Quadro 1.

Quadro 1: Critérios para definição da questão de pesquisa

População	Sistemas de TI voltados para o gerenciamento de espaços de estacionamento.
Intervenção	Utilização de sistemas computacionais para reduzir o tempo de procura de motoristas por vagas de estacionamento.
Resultados	Otimização do uso de vagas de estacionamento tendo em vista a redução do tempo de procura de estacionamento por motoristas e consequentemente a redução da emissão de carbono na atmosfera e do congestionamento nas cidades como solução para o desenvolvimento das cidades inteligentes.

Com o escopo da revisão definido foram criadas as seguintes questões de pesquisa.

1. Quais as pesquisas existentes que tratam do gerenciamento de estacionamentos no âmbito das cidades inteligentes?
2. Quais características relevantes estes trabalhos apresentam?

3.1.3 Critérios de Inclusão e Exclusão

De acordo com as questões de pesquisa e com o objetivo da revisão foram definidos Critérios de Inclusão e Exclusão relevantes para o tema da pesquisa com o objetivo de nortear a seleção dos artigos na fase de condução da revisão. Foram elaborados 5 Critérios de Inclusão (IC) e 5 Critérios de Exclusão (EC) como apresentados abaixo.

Os Critérios de Inclusão considerados na seleção dos artigos:

- **IC1.** O artigo apresenta algum algoritmo para gerenciar a ocupação de vagas de estacionamento.
- **IC2.** O artigo é sobre uma aplicação mobile/arquitetura/middleware dedicada à gestão de vagas de estacionamentos.
- **IC3.** O artigo propõe a criação de um aplicativo e possui descrição detalhada do funcionamento do mesmo.
- **IC4.** O artigo apresenta uma aplicação que utiliza sensores do próprio celular para definir as variáveis necessárias na implementação do estacionamento inteligente.
- **IC5.** O artigo possui informações pertinentes ao escopo do trabalho a exemplo pesquisas de interesse e dados relacionados a oferta e procura de estacionamentos.

Os Critérios de Exclusão considerados na seleção dos artigos:

- **EC1.** No artigo a temática da gestão do tráfego de veículos em cidades não é abordada.
- **EC2.** O estudo apresenta como solução exclusivamente sistemas embarcados.
- **EC3.** O estudo não propõe a criação de um aplicativo e não se trata de um algoritmo relacionado com o contexto.
- **EC4.** O artigo não está disponível para download, de forma completa, na Web.
- **EC5.** O trabalho não se atém ao escopo do tema, embora apresente um contexto similar não se aplica necessariamente ao desenvolvimento de aplicações mobile para gerenciamento de estacionamentos em cidades inteligentes.

3.1.4 String de Busca

Como etapa final do planejamento da revisão foi definida uma *string* de busca. Para este fim foram consideradas as palavras chaves do assunto de interesse e seus sinônimos como apresentados no Quadro 2.

Quadro 2: Termos e Sinônimos em Inglês

Termos	Sinônimos
System	Application Software
Parking	Car Park Parking Garage Parking lot Parking spot
Smart Cities	-

Com os termos da pesquisa definidos e utilizando os operadores booleanos <AND> (E) e <OR> (OU) foi elaborada uma *string* genérica de busca (Quadro 3), a qual foi utilizada nas bases bibliográficas durante a seleção dos artigos.

Quadro 3: String generica de busca

(Application OR Software OR System)AND("car park"OR"parking garage"
OR "parking lot"OR "parking spot"OR parking) AND ("smart cities")

3.1.5 Seleção dos estudos primários

Na fase de seleção dos estudos primários foram escolhidas 5 bases bibliográficas, nas quais foi realizada a busca por estes trabalhos, esta atividade foi realizada no mês de dezembro de 2016 e os resultados retornados correspondem ao que tinha disponível até esta data. As bases escolhidas são as listadas abaixo:

- Scopus ¹
- Science Direct ²
- Web of Science ³
- IEEE ⁴
- Engineering Village (Compendex) ⁵

Utilizando a *string* genérica e filtros nas diferentes bases, foram coletados os artigos de interesse para o trabalho. Determinou-se um período limite em algumas bases para refinar os resultados de forma que fossem retornados os artigos mais recentes, para isso foi definido um filtro de data a partir de anos maiores que 2005 para a Science Direct, de 2006 a 2016 para a Web of Science, de 2011 a 2016 para a IEEE e de 2007 a 2017 para a Engineering Village, não foi preciso utilizar um filtro de data para a Scopus, pois a mesma retornou artigos de anos recentes. As *strings* geradas em cada base de acordo com as entradas utilizadas são as especificadas no Quadro 4.

¹ Scopus: <https://www-scopus-com.ez20.periodicos.capes.gov.br/home.uri>

² ScienceDirect: <http://www-sciencedirect-com.ez20.periodicos.capes.gov.br/>

³ Web of Science: <https://goo.gl/1ukO2v>

⁴ IEEE Xplore: <http://ieeexplore.ieee.org.ez20.periodicos.capes.gov.br/Xplore/guesthome.jsp?reload=true>

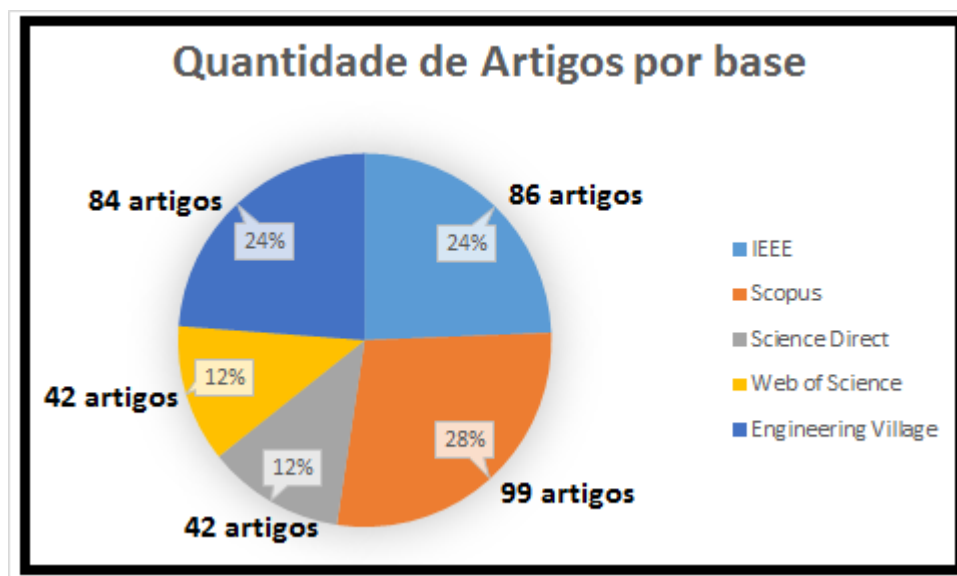
⁵ Engineering Village: <https://www-engineeringvillage-com.ez20.periodicos.capes.gov.br/search/quick.url>

Quadro 4: *String* de busca de cada base

Base	String Específica
Scopus	TITLE-ABS-KEY ((application OR software OR system OR suite) AND ("car park"OR "parking garage"OR "parking lot"OR "parking spot"OR parking) AND ("smart cities")) AND (LIMIT-TO (DOCTYPE , "cp") OR LIMIT-TO (DOCTYPE , "ar") OR LIMIT-TO (DOCTYPE , "ip"))
Science Direct	pub-date > 2005 and (Application OR Software OR System) AND ("car park"OR "parking garage"OR "parking lot"OR "parking spot"OR parking) AND ("smart cities") [All Sources(Computer Science)]
Web of Science	((Application OR Software OR System) AND ("car park"OR "parking garage"OR "parking lot"OR "parking spot"OR parking) AND ("smart cities"))Tempo estipulado: 2006-2016.Índices: SCI-EXPANDED, SSCI, A&HCI, CPCI-S, CPCI-SSH, ESCI.
IEEE	((Application OR Software OR System) AND ("car park"OR "parking garage"OR "parking lot"OR "parking spot"OR parking) AND ("smart cities")) and refined by Year: 2011 - 2016
Engineering Village	for 2007-2017: (Application OR Software OR System) AND ("car park"OR "parking garage"OR "parking lot"OR "parking spot"OR parking) AND ("smart cities")

Dessa forma a distribuição de artigos por base ficou como apresentado na Figura 3, sendo a Scopus a base com o maior número de resultados, seguido da IEEE e Engineering Village e com os menores resultados o Science Direct e Web of science , ambos com 42 dos artigos.

Figura 3 – Quantidade de Artigos coletados por base



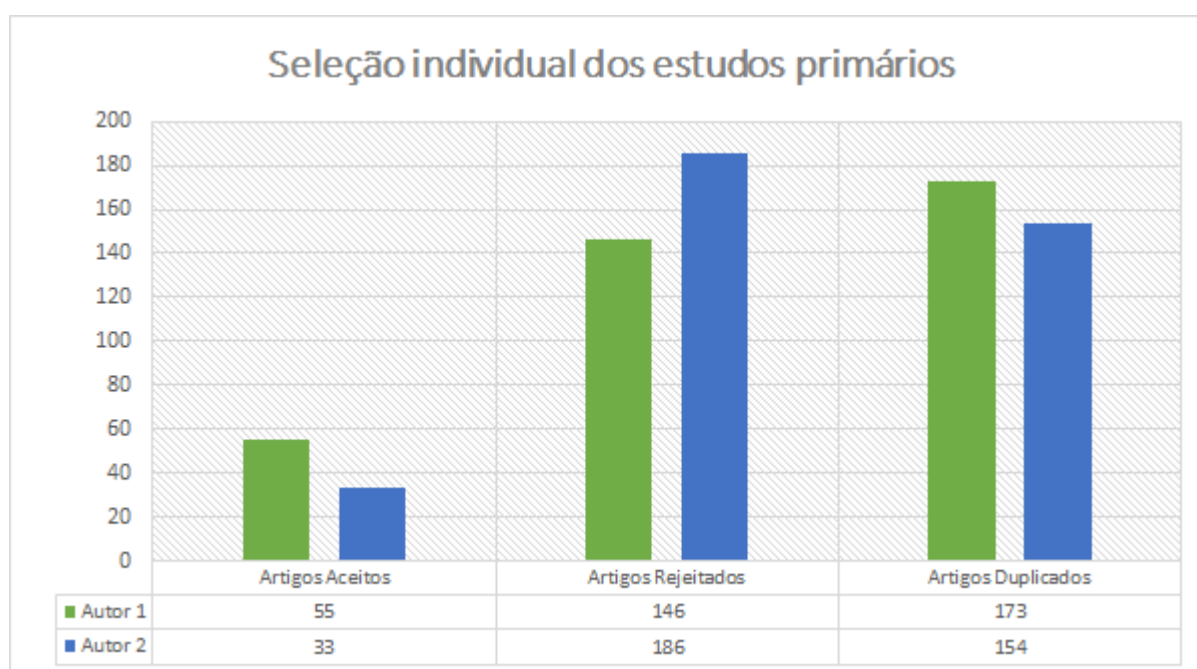
Fonte: Próprios autores

O processo de seleção se caracterizou em 3 etapas. Inicialmente foram baixados os arquivos bibtex⁶ de todos os artigos retornados pelas bases por meio da *string* de busca inserida.

⁶ O termo BibTeX se refere a uma ferramenta e um formato de arquivo que são usados para descrever e processar listas de referências, principalmente em conjunto com documentos LaTeX.

Com este arquivo foi utilizado o software StArt ⁷ para inserir as informações dos artigos e começar o processo de seleção. O trabalho aqui apresentado foi realizado por dois autores, ambos alunos do curso de Bacharelado em Sistemas de Informação na Universidade Federal de Sergipe (UFS), e no processo de seleção estes autores trabalharam de forma separada, ou seja, tanto o Autor 1 como o Autor 2, de forma separada, leram os títulos, resumos e palavras-chave para então escolherem os artigos que foram relevantes de acordo com os critérios. Nesta fase foram eliminadas as divergências e os artigos duplicados. A Figura 4 apresenta os resultados desta fase.

Figura 4 – Artigos selecionados, rejeitados e duplicados durante seleção individual - Autor 1 e Autor 2.



Fonte: Próprios autores

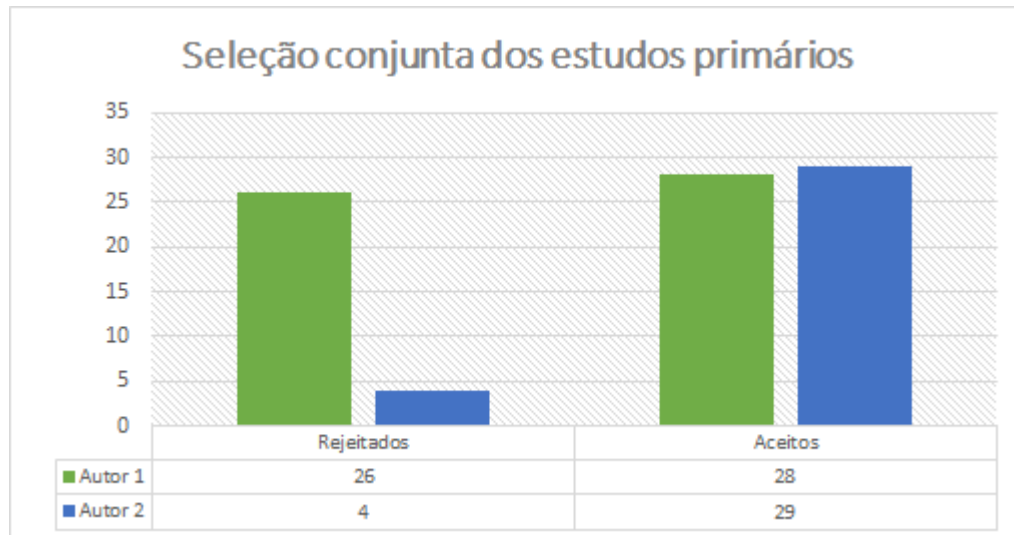
Uma vez que esta pré seleção foi realizada, os autores se reuniram para comparar suas escolhas e fazer a seleção final dos artigos. Nesta fase, foram aceitos todos os artigos que ambos escolheram, 25 no total, e identificado que 1 dos artigos escolhidos pelo Autor 1 era duplicado. Os trabalhos restantes foram reavaliados e foi discutida a sua relevância. Após a discussão, mais 2 artigos da lista de escolhidos do Autor 1 foram aceitos e 11 artigos, sendo 6 do Autor 1 e 5 do Autor 2, foram escolhidos para serem analisados mais profundamente, explorando mais do seu conteúdo.

Destes 11 artigos, 5 foram selecionados, sendo 1 do Autor 1 e 4 do Autor 2, totalizando 28 artigos selecionados da lista do Autor 1 e 29 artigos selecionados da lista do Autor 2. Dos 25 artigos escolhidos por ambos não foi possível baixar 3 dos artigos, pois a base Scopus não estava permitindo o download do artigo completo apenas os resumos, sendo assim esses artigos

⁷ Download do StArt em http://lapes.dc.ufscar.br/tools/start_tool

que eram da base Scopus e que não foram encontrados em nenhuma outra base foram rejeitados nesta fase de seleção.

Figura 5 – Artigos aceitos e rejeitados na fase de seleção conjunta



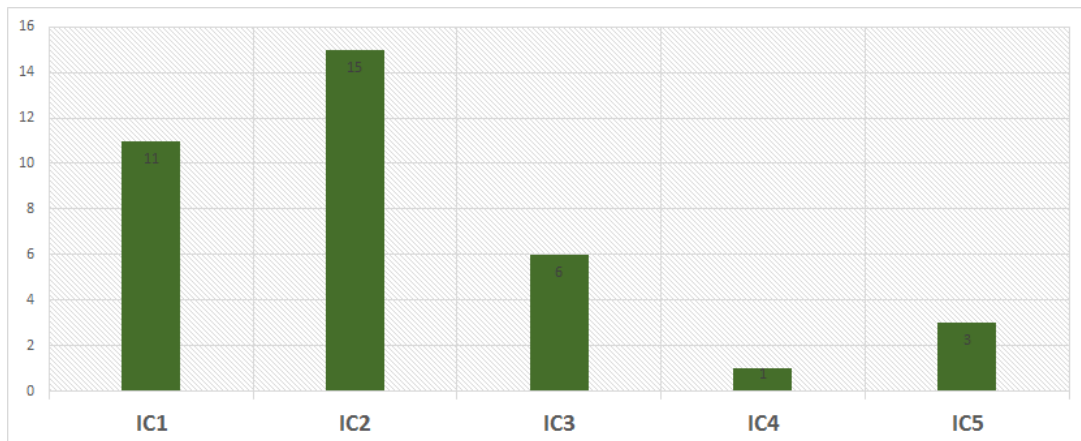
Fonte: Próprios autores

Na fase seguinte os 29 artigos restantes foram divididos entre os autores para a extração das características de interesse para o trabalho. A Figura 5 mostra a quantidade total de artigos selecionados e rejeitados das listas dos Autores 1 e 2, sendo que dos 55 artigos selecionados pelo Autor 1, 1 deles foi identificado como duplicado, 26 foram rejeitados e 28 foram aceitos na seleção conjunta. E dos 33 artigos escolhidos pelo Autor 2, 4 foram rejeitados e 29 foram aceitos na seleção conjunta.

3.1.6 Extração

Na fase de extração notou-se que 1 dos 29 artigos possuía o mesmo conteúdo que um outro que também foi selecionado e este foi eliminado desta fase restando 28 artigos para extração das características. Os 28 artigos foram selecionados de acordo com os critérios de inclusão apresentados na Figura 6.

Figura 6 – Quantidade de artigos por critério de inclusão



Fonte: Próprios autores

Após a leitura por completo de cada artigo foi elaborada uma síntese com os pontos principais que cada um apresentava com o objetivo de responder a primeira questão de pesquisa elaborada: *Quais as pesquisas existentes que tratam do gerenciamento de estacionamentos no âmbito das cidades inteligentes?*

O Quadro 5 apresenta a relação do título do artigo, seus autores e um respectivo identificador.

Quadro 5: Identificação dos Estudos

ID	Título	Autores
A1	IoT based smart parking system	(KHANNA; ANAND, 2016)
A2	Reservation-based multi-objective smart parking approach for smart cities	(MEJRI et al., 2016)
A3	A wireless smart parking system	(ORRIE; SILVA; HANCKE, 2015)
A4	DisAssist: An internet of things and mobile communications platform for disabled parking space management	(LAMBRINOS; DOSIS, 2013b)
A5	Smart parking: Parking occupancy monitoring and visualization system for smart cities	(GRODI; RAWAT; RIOS-GUTIERREZ, 2016)
A6	Prediction system of occupancy of parking spaces	(SROTYR et al., 2015)
A7	Parking availability prediction for sensor-enabled car parks in smart cities	(ZHENG; RAJASEGARAR; LECKIE, 2015)
A8	A cloud-based intelligent car parking services for smart cities	(JI et al., 2014b)
A9	A probabilistic approach to parking: Benefits of routing instead of spotting	(FEHER; LAJTHA; LOVASZ, 2015)

(Continuação)

ID	Título	Autores
A10	Smart parking solutions for urban areas	(POLYCARPOU; LAMBRINOS; PROTOPAPADAKIS, 2013)
A11	City of White Plains Parking App: Case Study of a Smart City Web Application	(BUTOWSKY et al., 2015)
A12	Optimal Allocation of Public Parking Slots Using Evolutionary Algorithms	(ARELLANO-VERDEJO; ALBA, 2016)
A13	Unlocking the smartphone's senses for smart city parking	(KRIEG et al., 2016)
A14	A Smart Parking System based on IoT protocols and emerging enabling technologies	(MAINETTI et al., 2015)
A15	Smart traffic parking management using IoT	(ROY et al., 2016)
A16	Secure Smart Parking at James Madison University via the Cloud Environment (SPACE)	(GARCIA et al., 2016)
A17	BlueParking: An IoT based parking reservation service for smart cities	(TAHERKHANI et al., 2016)
A18	A Real-Time Parking Prediction System for Smart Cities	(VLAHOGIANNI et al., 2016)
A19	An Architecture for a Mobility Recommender System in Smart Cities	(MARTINO; ROSSI, 2016)
A20	A new heuristic for solving the parking assignment problem	(ABIDI et al., 2015)
A21	A cloud-based car parking middleware for IoT-based smart cities: Design and implementation	(JI et al., 2014a)
A22	A social-aware smart parking application	(DI NOCERA; DI NAPOLI; ROSSI, 2015)
A23	Applying mobile and internet of things technologies in managing parking spaces for people with disabilities	(LAMBRINOS; DOSIS, 2013a)
A24	Simulation environment for investigating crowd-sensing based urban parking	(FARKAS; LENDÁK, 2015)
A25	Constructing Execution and Life-Cycle Models for Smart City Services with Self-Aware IoT	(NAKAMURA; BOUSQUET, 2015)
A26	Internet of Things Approach to Cloud-based Smart Car Parking	(ATIF; DING; JEUSFELD, 2016)
A27	Smart parking by mobile crowdsensing	(CHEN; LIU, 2016)
A28	Mitigating traffic problems by integrating smart parking solutions into an interconnected ecosystem	(STRASSER; MAUSER; ALBAYRAK, 2016)

O Quadro 6 apresenta a síntese de cada artigo, na qual foram retirados os pontos principais abordados no trabalho.

Quadro 6: Síntese dos pontos mais relevantes extraídos nos estudos

Fonte	Síntese
A1	Khanna e Anand (2016) propõem um estacionamento inteligente, baseado na integração entre dispositivos conectados (IoT) e interagindo com uma aplicação na nuvem (Cloud Computing). A arquitetura tem como componentes: sensores de estacionamento, unidades de processamento (que fazem um processamento inicial dos dados oriundos dos sensores), uma aplicação na nuvem (que recebe os dados das unidades de processamento e realiza processamentos adicionais) e uma aplicação mobile (que é utilizada pelos motoristas para interagir com a aplicação que executa na nuvem).
A2	Mejri et al. (2016) propõem um sistema de reserva inteligente de vagas de estacionamento, capaz de lidar com múltiplos objetivos. O sistema, denominado ROSAP, procura equilibrar três fatores: a distância do estacionamento em relação a posição atual do motorista no momento da requisição, a distância entre o estacionamento e o destino final do motorista (o quanto ele tem que caminhar partindo do estacionamento até o seu destino final), e o impacto daquela requisição no tráfego em geral. Para isso foi proposta uma meta-heurística de arrefecimento simulado. Ainda é apresentada uma simulação comparando o desempenho do algoritmo proposto com um algoritmo guloso (tipicamente utilizado nesse tipo de problema). O ROSAP se mostrou superior à abordagem gulosa, em quase todos os critérios avaliados, incluindo uma superioridade na taxa de satisfação em relação à abordagem gulosa de 60% com tráfego normal e de 35% com tráfego intenso.
A3	Orrie, Silva e Hancke (2015) propõem um sistema sem fio para localização de vagas de estacionamento que para tal faz uso de uma rede de sensores sem fio que verificam se um local de estacionamento está ocupado ou vago. O sistema proposto consiste de uma aplicação para <i>smartphones</i> e uma rede de sensores <i>wireless</i> .
A4	Lambrinos e Dosis (2013b) propõem o DisAssist, um sistema que integra sensores e <i>smartphones</i> , com o propósito de prover uma melhor gestão das vagas de estacionamento reservadas para pessoas com deficiência. Funciona de forma que sensores capturam a disponibilidade de vagas e envia os dados para o servidor, por meio de um gerenciador de reservas as novas requisições de reserva são geridas e monitoradas, Quando um usuário chega a vaga reservada, este passa por uma verificação para comprovar que possui alguma necessidade especial, se a autenticação é confirmada a vaga é marcada como ocupada legitimamente, se não a vaga é marcada como ocupada ilegitimamente e as informações sobre a violação são enviadas às autoridades competentes.
A5	Neste artigo, Grodi, Rawat e Rios-Gutierrez (2016) apresentam um protótipo de sistema de estacionamento inteligente utilizando a tecnologia de redes de sensores sem fio. Por meio desta rede, o status do local de estacionamento (ocupados ou livres) são detectados e transmitidos a um banco de dados. Essas informações podem ser acessadas pelos usuários por meio do site ou do aplicativo móvel desenvolvidos, que recebem atualizações dos sensores em tempo real. Este sistema fornecer aos usuários atualizações quase instantâneas dos locais de estacionamento disponíveis ao passo que a WSN permite flexibilidade no posicionamento dos sensores.

(Continuação)

Fonte	Síntese
A6	Srotyr et al. (2015) propõem o desenvolvimento de um sistema Telemático, que recebe como entrada as informações capturadas por um sistema de pedágios e com base nestas informações utiliza um modelo de previsão para prever a ocupação das vagas de estacionamento nas rodovias com o objetivo de disponibilizar estas informações para os motoristas e otimizar o uso destas vagas, principalmente pelos carros pesados. Esta aplicação é focada nas rodovias da República Checa e é dividida em três subsistemas: Um subsistema de banco de dados, um subsistema de predição e um subsistema de distribuição das informações. O protótipo de um aplicativo foi desenvolvido e para avaliar o modelo de previsão, foram realizados testes na Rodovia D5 deste país. Nestes testes foram comparadas as previsões do modelo com o status real de ocupação dos espaços de estacionamento e obteve-se uma taxa de 84% de acertos, segundo os autores.
A7	Zheng, Rajasegarar e Leckie (2015) propõem um estudo sobre diferentes algoritmos para definir a taxa de ocupação de vagas de estacionamento usando 3 diferentes conjuntos de dados, dados estes providos de bases de duas diferentes cidades, a cidade de São Francisco (EUA) e Melbourne (Austrália), as quais possuem sensores instalados sob a estrada nos locais de estacionamento. O estudo destes algoritmos tem por objetivo definir uma forma de prever quando uma vaga de estacionamento estará ocupada. Foram analisados os algoritmos da Árvore de Regressão, Regressão do Vetor de Suporte e Rede Neural e concluiu que a Árvore de regressão é o melhor algoritmo de predição depois de comparado com os outros dois algoritmos. E o uso de dados históricos aumenta a precisão dos resultados.
A8	Ji et al. (2014b) apresentam uma infraestrutura baseada em nuvem dividida em três camadas, aplicação, comunicações e sensores, a qual oferece serviços de locação de estacionamento, informações sobre estacionamentos, supervisão de estacionamento, GPS, patrulhamento de placas de carro e rastreamento. A arquitetura é formada por uma estação central, a qual recolhe informações de sensores distribuídos ao redor do estacionamento e armazenamento noSQL. Foi construída uma aplicação específica para ser usada em uma universidade que funciona de forma que o quando motorista chega próximo à universidade uma requisição http automática é enviada ao <i>web service</i> pedindo informações de vagas disponíveis e uma resposta JSON é retornada. A aplicação de estacionamento é considerada uma das mais importantes em um Sistema de transporte inteligente.
A9	Feher, Lajtha e Lovasz (2015) propõem uma solução baseada em probabilidade, na qual são sugeridas rotas com maior probabilidade de possuírem vagas de estacionamento disponíveis. Pode acontecer de nenhuma rota ser encontrada, assim como muitas rotas podem ser encontradas, nesse caso a rota sugerida para o usuário é baseada em um cálculo que mede o custo da distância do estacionamento até o destino do motorista. O sistema fica mais preciso cada vez que as rotas sugeridas realmente possui uma vaga disponível e o motorista estaciona em uma vaga nesta rota. Para isso a aplicação possui um banco de dados de probabilidade, um banco de dados de eventos, um serviço de predição e um cliente móvel.

(Continuação)

Fonte	Síntese
A10	O estudo realizado por Polycarpou, Lambrinos e Protopapadakis (2013) tem como objetivo levantar informações sobre a opinião dos motoristas em relação ao uso das vagas de estacionamento. O trabalho levanta as características importantes de um estacionamento inteligente partindo do uso de sensores, seus tipos, a sistemas baseados em reservas e preços dinâmicos e apontam algumas questões a considerar sobre um aplicativo para este fim.
A11	A proposta de Butowsky et al. (2015) é fazer um estudo de caso na cidade de White Plains (EUA). Para isso são apresentadas três aplicações: um aplicativo que já funciona na cidade, uma aplicação web desenvolvida para o trabalho e uma perspectiva futura da Internet das Coisas. São apresentadas as vantagens de uma aplicação web sobre uma aplicação Android e descritas as ferramentas utilizadas para o desenvolvimento (HTML, CSS e JavaScript). O autor também comenta sobre a implementação de requisições HTTP mais leves, as quais demoram menos para carregar as páginas web.
A12	Arellano-Verdejo e Alba (2016) propõem um algoritmo para alocação ótima de vagas de estacionamento baseado no Algoritmo Evolutivo de Estado Estacionário. Para validar a eficiência do algoritmo proposto foram realizados testes em comparação a outros dois, o algoritmo de Pesquisa Randômica (<i>Random Search</i>) e a técnica de Arrefecimento Simulado (<i>Simulated Annealing</i>). As instâncias dos testes se basearam em três situações: uma na qual o número de vagas é menor do que o número de carros, outra na qual o número de vagas é igual ao número de carros e uma última na qual o número de vagas é maior que o número de carros. A partir das simulações realizadas percebeu-se que o algoritmo proposto é robusto o suficiente para resolver o problema de maneira mais eficiente em relação aos outros dois apresentados.
A13	Krieg et al. (2016) apresentam uma solução que visa atingir duas questões chaves: detectar o tipo de transporte que está sendo utilizado e identificar se o motorista voltou para o mesmo carro que estacionou. Para isso os autores utilizam os próprios sensores dos smartphones e um algoritmo de classificação baseado no <i>Random Forest</i> para classificar as leituras dos sensores e identificar o transporte usado. Para identificar a primeira e a segunda questão são utilizados os próprios sensores do smartfone. Exaustivos testes foram realizados nessa aplicação e o resultado em ambas as funcionalidades atingiu uma precisão de mais de 95%.
A14	Mainetti et al. (2015) apresentam uma infraestrutura de sensores e padrões de comunicação híbridos trabalhando em integração com uma plataforma em nuvem e dois aplicativos para smartfone, um para um motorista e outro para uma autoridade responsável pelo patrulhamento do trânsito, a fim de tornar este último apto a registrar e autuar infrações no que se trata da ocupação não autorizada de vagas de estacionamentos. O aplicativo do motorista funciona de forma que o usuário visualiza a ocupação das vagas de estacionamento em uma determinada região, recebe as direções corretas para chegar a esta vaga, checa o tempo restante de estacionamento e recebe notificações de quando o tempo contratado estiver acabando. O Sistema também permite pagamentos utilizando TAG NFC por meio de uma aplicação que foi desenvolvida pelos autores em outro trabalho. É utilizado um <i>raspberry PI</i> como <i>gateway</i> entre a rede de sensores híbridos e a plataforma em nuvem e esta última funciona como gerenciador da aplicação, função principal da aplicação proposta. Também é possível utilizar a aplicação via <i>web browser</i> .

(Continuação)

Fonte	Síntese
A15	O sistema proposto por Roy et al. (2016) apresenta 3 funções principais: controle de congestionamento, gerenciamento de vagas de estacionamento e um mecanismo para cobrança automática de imposto de pedágio. O controle de estacionamento funciona de forma que um dispositivo com o IBM Watson instalado fica em cada esquina atualizando o servidor central de tempo em tempos com informações do tráfego. Quando o motorista faz uma solicitação e informa o destino o sistema utiliza o algoritmo do caminho mais curto de Dijkstra e retorna para o motorista o melhor caminho a seguir. O gerenciamento das vagas de estacionamento, por sua vez, funciona com o auxílio de uma rede de sensores sem fio. O objetivo é informar aos motoristas sobre espaços de estacionamento disponíveis em áreas próximas. Por fim, uma central inteligente mantém um sistema de pagamento eletrônico que irá fazer uma dedução automática do cartão de crédito do motorista do carro que foi cadastrado no sistema.
A16	A aplicação apresentada por Garcia et al. (2016) é focada em administrar o estacionamento da James Madison University, que já possui um sistema que conta quantos carros entram e saem do estacionamento, mas não possui um mecanismo que informe quantas vagas estão disponíveis em determinado estacionamento naquele momento. É detectada a localização do motorista e exibida o número de espaços disponíveis, o qual é obtido de um banco de dados armazenado em nuvem. Para alcançar a vaga, os usuários interagem com um mapa para adquirir as direções para uma locação de estacionamento em particular. A aplicação também pode notificar motoristas da probabilidade de encontrar uma vaga de estacionamento, usando a localização do motorista, condições de tráfego e dados históricos uma probabilidade estimada de encontrar uma vaga no tempo de chegada ao destino será fornecido.
A17	Taherkhani et al. (2016) apresentam uma aplicação baseada em nuvem e que utiliza o conceito de Internet das Coisas para a reserva de vagas de estacionamento. A aplicação encontra eficientemente a melhor vaga de estacionamento disponível próxima ao destino do motorista, tendo como base um roteamento inteligente e a estimação do tempo da viagem, além de considerar as preferências do usuário (próximo ao destino, menor custo).
A18	Vlahogianni et al. (2016) apresentam o desenvolvimento de um modelo para prever a ocupação esperada em estacionamentos. Para análise dos dados utilizam a grande disponibilidade de dados, em tempo real, de estacionamentos obtidos pelos sensores espalhados pela cidade de Santander na Espanha. O objetivo deste trabalho é explorar métodos estatísticos e de inteligência computacional para o desenvolvimento de uma metodologia que possa ser usada para previsão da ocupação destas vagas. A metodologia é avaliada e é apresentado um sistema web explorando, em tempo real, os modelos de previsão propostos.
A19	Martino e Rossi (2016) descrevem uma arquitetura distribuída de dispositivos conectados utilizados para prover informações ao Sistema de Recomendação de Mobilidade (SRM). O diferencial desse sistema é que entre as rotas sugeridas ao motorista estão não apenas as que apresentam um local de estacionamento mais próximo do destino final, mas, também rotas que consideram trocas do meio de transporte ao longo do percurso, que são rotas nas quais o motorista estaciona seu veículo em um ponto do caminho e segue viagem com outro meio de transporte (ônibus, trem, metrô, etc.) até chegar ao seu destino final.

(Continuação)

Fonte	Síntese
A20	No artigo, Abidi et al. (2015) , é apresentado o problema da atribuição de vagas de estacionamento livre considerando restrições de tempo e, para solucioná-lo, apresentam um algoritmo genético híbrido, o <i>Hybrid Genetic Assignment Search Procedure (HGASP)</i> , que combina um algoritmo genético com um algoritmo de procedimento de busca aleatório e adaptativo. Em seguida, realizaram uma simulação onde compararam o desempenho do HGASP com outros algoritmos: SSGA, GRASP e o algoritmo aleatório. O HGASP se saiu melhor em todos os cenários avaliados (embora seus resultados tenham sido próximos ao do SSGA).
A21	Ji et al. (2014a) apresentam um conceito genérico de usar serviços de estacionamento inteligentes baseados na computação em nuvem em cidades inteligentes como uma importante aplicação do paradigma da Internet das Coisas (IoT). Estes tipos de serviços se tornarão parte integrante de uma plataforma operacional IoT genérica para Cidades Inteligentes devido aos seus recursos. Uma visão de alto nível do <i>middleware</i> proposto é delineada e a plataforma operacional correspondente é ilustrada. Para demonstrar a oferta de serviços de estacionamento de carros, com base no <i>middleware</i> proposto, é descrito um sistema de estacionamento inteligente baseado em nuvem para uso dentro de um campus universitário, juntamente com detalhes de seu projeto, implementação e operação. Uma série de soluções de software, incluindo os <i>clusters</i> Kafka / Storm / Hbase, aplicações web OSGi com NoSQL distribuído, um mecanismo de regras e aplicações móveis, são propostas para oferecer a melhor experiência de serviço de estacionamento a usuários móveis, seguindo o paradigma <i>Always Best Connected and best Served</i> (ABC & S).
A22	Di Nocera; Di Napoli e Rossi (2015) apresentam um protótipo de uma aplicação web para estacionamento inteligente, baseado em uma abordagem de negociação e realizam uma simulação para avaliar a adequação do mecanismo proposto como uma solução de estacionamento inteligente. Na abordagem proposta, a decisão sobre onde estacionar é o resultado de um processo automatizado de negociação entre dois agentes de software: o Agente de Usuário (AU) agindo em nome do motorista, e o Gerenciador de Estacionamento (GE) que é responsável pela gestão dos espaços de estacionamento localizados na cidade que são oferecidos aos usuários. Os resultados da simulação apontam que as negociações promovidas pelo software proposto contribuem para um maior benefício para a cidade como um todo, contudo esse diferença é pequena quando comparado com as simulações onde não ocorrem as negociações e, apenas o interesse dos motoristas, e não o impacto na cidade como um todo, é levado em conta.

(Continuação)

Fonte	Síntese
A23	O estacionamento em áreas públicas é um dos principais problemas enfrentados nos ambientes urbanos modernos. Esse é um problema ainda maior no caso dos cidadãos com deficiência que têm um número limitado de espaços atribuídos para o seu uso exclusivo, que muitas vezes não são suficientes para satisfazer a demanda e, por vezes, são ilegitimamente ocupados. Um sistema para Cidades Inteligentes que combina comunicação móvel e de máquina, sendo que a máquina foi concebido por Lambrinos e Dosis (2013a) com o objetivo de aliviar o problema descrito. Utiliza sensores para adquirir informações da disponibilidade de locais de estacionamento para pessoas com deficiência, que são divulgados em tempo real aos usuários registrados. Utilizando essas informações, os usuários podem dirigir diretamente para pontos atualmente disponíveis ou até mesmo reservar um deles. Após a sua chegada, os usuários legítimos são capazes de confirmar sua identidade por meio de uma mensagem de texto simples(SMS), um dispositivo especial ou de um aplicativo em seus <i>smartphones</i> . A autenticação dos usuários acaba por permitir um monitoramento mais eficiente desses locais de estacionamento.
A24	Farkas e Lendák (2015) ressaltam a importância de utilizar simulações para análise do comportamento de uma aplicação no contexto do <i>crowd-sensing</i> , pessoas atuando como sensores por meio do compartilhamento de informações, tendo em vista que utilizando simulações é possível ter uma base de quais valores são necessários para a aplicação ter sucesso e qual o impacto da mesma. A fim de enfatizar a proposta é construído um cenário que tem como foco principal analisar o comportamento dos motoristas dirigindo seus carros em áreas urbanas na procura por vagas para estacionar próximas a seu destino. A simulação é influenciada pela interação dos motoristas compartilhando informações sobre eventos relacionados ao estacionamento (ocupação ou desocupação de vaga). Embora o cenário considerado tenha sido limitado o trabalho trouxe resultados interessantes no que diz respeito a redução do tempo gasto na procura de vaga para estacionar em virtude de estacionamentos lotados que diminuiu em 14%.
A25	Nakamura e Bousquet (2015) propõem criar um modelo baseado no conceito de cidade como uma máquina de estados, apresentando um modelo de execução do serviço e um modelo do ciclo de vida. O conceito de cidade como máquina de estados considera que a cidade tem um estado, o qual pode ser bom ou ruim, dependendo do ponto de vista do serviço. Executando uma ação na cidade é possível mudar seu estado até chegar-se no estado considerado ideal. O modelo é aplicado em quatro serviços de estacionamento inteligente em quatro diferentes cidades adaptando-se cada serviço dentro do modelo proposto.
A26	Atif, Ding e Jeusfeld (2016) mostram a agenda de um trabalho em progresso que visa transformar o gerenciamento de vagas de estacionamento em um modelo de negócio que transforma o estacionamento em um serviço computacional a fim de proporcionar o melhor uso dos espaços de estacionamento.

(Continuação)

Fonte	Síntese
A27	Chen e Liu (2016) apresentam o projeto de um sistema de estacionamento inteligente baseado em <i>crowdsensing</i> , enquanto um estudo de caso específico, numa tentativa de explorar os princípios básicos de projeto aplicáveis a uma série de aplicações semelhantes. Por meio de simulações, é demonstrada que as estratégias por trás das atividades de <i>crowdsensing</i> podem influenciar a utilidade de tais aplicativos de forma significativa. Igualmente importante, é a constatação que um certo nível de "freeriding" (não determinismo) pode ser permitido para aumentar os benefícios sociais, desde que exista um mecanismo razoável de diferenciação de serviços. Os resultados obtidos dão aos projetistas de software uma melhor compreensão das funcionalidades móveis do <i>crowdsensing</i> e ajudam a orientar projetos para que sejam bem sucedidos.
A28	Strasser, Mauser e Albayrak (2016) propõem integrar mercados fechados de serviços inteligentes em um único ecossistema de serviços, capaz de transcender os limites do mercado individual e possibilitar ao usuário uma abrangência maior de serviços, sem precisar de várias contas de acesso em diferentes mercados para isto. No contexto das Cidades Inteligentes, significa dizer que um usuário pode oferecer e consumir um serviço que for do seu interesse. Para provar que a arquitetura é praticável é utilizada a oferta de serviços de vagas de estacionamento.

Também buscando responder a segunda questão de pesquisa, as características principais identificadas nos estudos selecionados estão descritas do Quadro 7.

Quais características relevantes estes trabalhos apresentam?

Quadro 7: Identificação das características

Característica	Fonte
Utiliza algoritmos ligados à inteligência artificial	A2, A6, A7, A9, A12, A18, A19, A20, A22
Utiliza rede de sensores	A1, A3, A4, A5, A15, A16, A19, A23
Possui aplicação Web	A5, A11, A14, A16, A22
Apresenta um Aplicativo como proposta	A1, A2, A3, A4, A5, A6, A8, A9, A13, A14, A16, A17, A19, A21, A23
Utiliza os sensores do próprio smartphone	A13, A14
Baseados em <i>crowd-sensing</i>	A24, A27
Utiliza simulações para validar os resultados	A2, A12, A20, A24, A27
Baseado em nuvem	A1, A2, A3, A4, A5, A8, A14, A16, A17, A21, A26
Utiliza o conceito de reservas	A1, A2, A4, A17
Discute resultados de um <i>survey</i>	A10

O levantamento das características dos trabalhos explorados na Revisão Sistemática foi de suma importância no norteamto do planejamento da solução apresentada neste trabalho, uma vez que trouxe conceitos que serviram como ponto de partida na elaboração das funcionalidades

da solução como a utilização de *crowd-sensing*, a incorporação da função de reserva de vagas e a utilização de serviços em nuvem.

3.2 Revisão de Produtos no Mercado

Não apenas os artigos foram analisados, como também as ferramentas existentes no mercado que auxiliam no gerenciamento das vagas de estacionamento. Um processo de seleção semelhante ao dos artigos aconteceu na Revisão de Produtos no Mercado, na qual a partir de diferentes strings de busca (apresentadas nos Quadros 9, 10, 11, 12, 13, 14, 15 e 16) inseridas na Play Store⁸ e na ferramenta de pesquisa da Google⁹ uma lista de aplicativos foi selecionada. Dos 50 aplicativos que foram retornados nas buscas, 16 foram escolhidos para uma análise mais aprofundada e por fim 8 foram tomados como referência para o produto de software desenvolvido neste trabalho.

O Quadro 8 apresenta os aplicativos analisados e os URLs para acesso a estes.

Quadro 8: Aplicativos e URLs para acesso

Aplicativo	Url
Parking: Localizar meu carro	https://goo.gl/KZH7ds
ParAli Estacionamento	https://goo.gl/pRhlfv
ParkApp Admin Estac. e Valets	https://goo.gl/gHtwkA
Himachal Parking	https://goo.gl/Hi3Vuw
Onde Parar	https://goo.gl/KbHl7f
TelPark	https://goo.gl/74JK5M
ParkingAki	https://goo.gl/QRDvG6
ParkMe	https://goo.gl/5JvdR5
Smart Parking	https://marvelapp.com/544j334

As seções abaixo descrevem o que foi encontrado na exploração destes aplicativos, apresentando algumas de suas telas e como cada um deles funciona.

3.2.1 Parking: Localizar meu carro

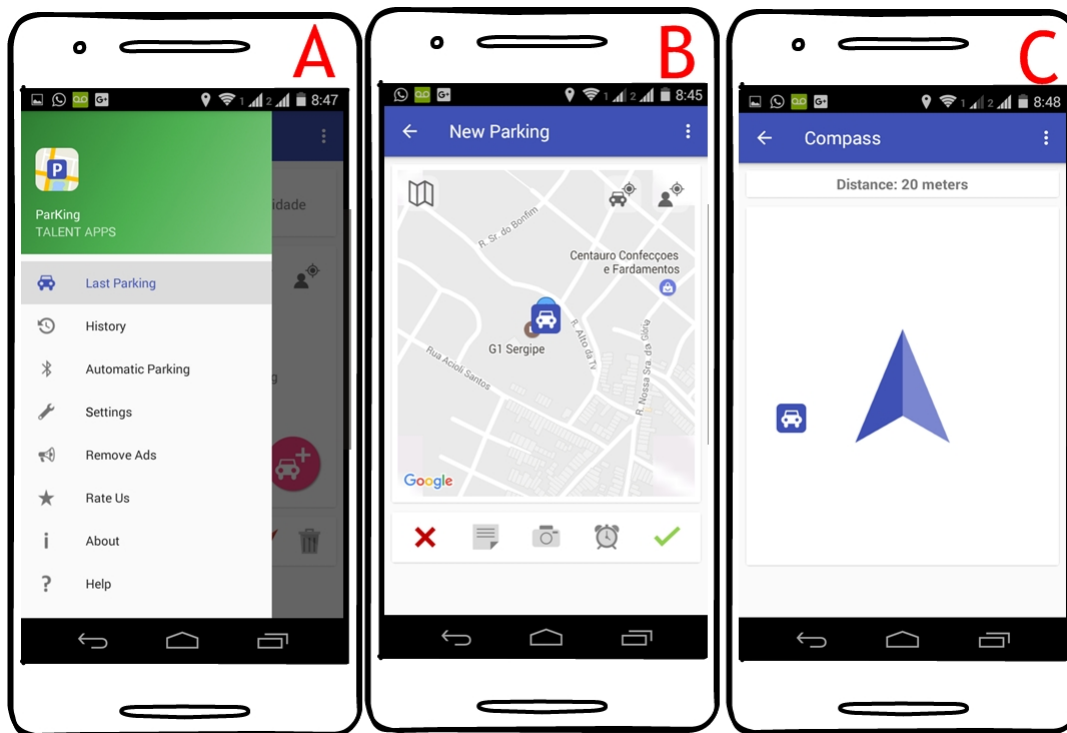
O aplicativo possui alta inteligibilidade em sua interface que possui seis opções: "Estacionar"(onde o usuário marca o local que estacionou, tira foto do lugar, escreve alguma nota que ache pertinente); "Estacionamento Automático", pelo app, o usuário ativa o *bluetooth*, e assim que se afastar do dispositivo pareado (o *bluetooth* do veículo, por exemplo) a localização do carro será marcada e o veículo será dado como estacionado; "Histórico"(que armazena os locais onde o veículo foi estacionado), "Configurações"(que permite escolher o idioma, qual tipo de mapa será exibido, qual unidade de distância será utilizada, etc.); "Sem anúncios"(direciona

⁸ Apps para Android no Google Play: <https://play.google.com/store/apps>

⁹ Google: <https://www.google.com>

para uma versão *premium* do aplicativo que não exibe anúncios, mas não é gratuita); "Sobre nós"(exibe os contatos da empresa desenvolvedora). O aplicativo (Figura 7) não possui muitas funcionalidades, mas é prático de usar e consegue ser útil em ajudar os motoristas a localizarem seus veículos.

Figura 7 – Telas do aplicativo Parking: Localizar meu Carro



Fonte: Adaptado de [TalentApps \(2017\)](#)

A Figura 7.A apresenta a tela de menu do aplicativo, na qual são exibidas todas as opções que este oferece, seguido da tela de novo estacionamento (Figura 7.B), na qual o usuário informa que estacionou e uma bússola, a qual é usada para orientar o motorista para a localização do seu carro (Figura 7.C). Este aplicativo foi encontrado a partir da *string* de busca exibida no Quadro 9.

Quadro 9: String de busca utilizada na pesquisa do Parking: Localizar meu carro

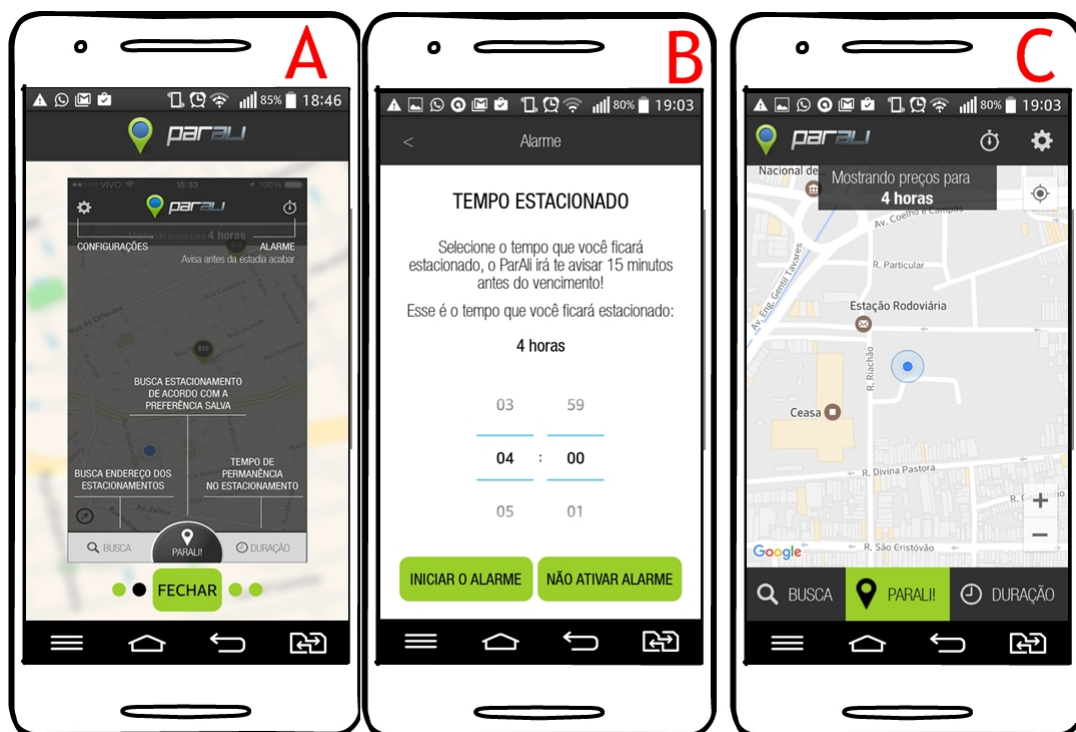
String: *parking* + Preço: Gratuitos + Classificação: Quatro Estrelas ou Mais + Categoria: Mapas e Navegação

Por meio da palavra *parking* e os filtros para aplicativos gratuitos que possuem quatro estrelas ou mais e que estão na categoria de Mapas e Navegação uma lista de aplicativos que se enquadravam nestes filtros foram retornados entre eles o Parking : Localizar meu carro.

3.2.2 ParAli Estacionamento

O aplicativo possui aspectos de usabilidade como apreensibilidade e atratividade. Na tela inicial do aplicativo o usuário enxerga um mapa com sua localização atual. No canto inferior ele pode selecionar um local de destino (para onde será redirecionado e o mapa exibirá os estacionamentos na região), é possível selecionar quanto tempo deseja permanecer no estacionamento (os preços exibidos variam de acordo com o tempo selecionado), no canto superior esquerdo é possível acessar as configurações, onde o usuário pode selecionar suas preferências: qual deve ser o filtro da busca de estacionamentos (o mais perto, o mais barato, meio termo), qual o tamanho do seu veículo, etc. A Figura 8 apresenta algumas telas deste aplicativo.

Figura 8 – Telas do aplicativo ParAli Estacionamento



Fonte: Adaptado de ParAli (2014)

A Figura 8.A exibe a tela de apresentação do aplicativo, a qual exibe um tutorial para o uso das funcionalidades oferecidas, seguido da tela de configuração do tempo de estacionamento (Figura 8.B) e a tela com a localização atual (Figura 8.C), na qual são exibidas os estacionamentos próximos. Na busca por este aplicativo a *string* apresentada no Quadro 10 foi utilizada.

Quadro 10: String de busca utilizada na pesquisa do ParAli Estacionamento

String: *parking* + Preço: Gratuitos + Classificação: Quatro Estrelas ou Mais + Categoria: Mapas e Navegação

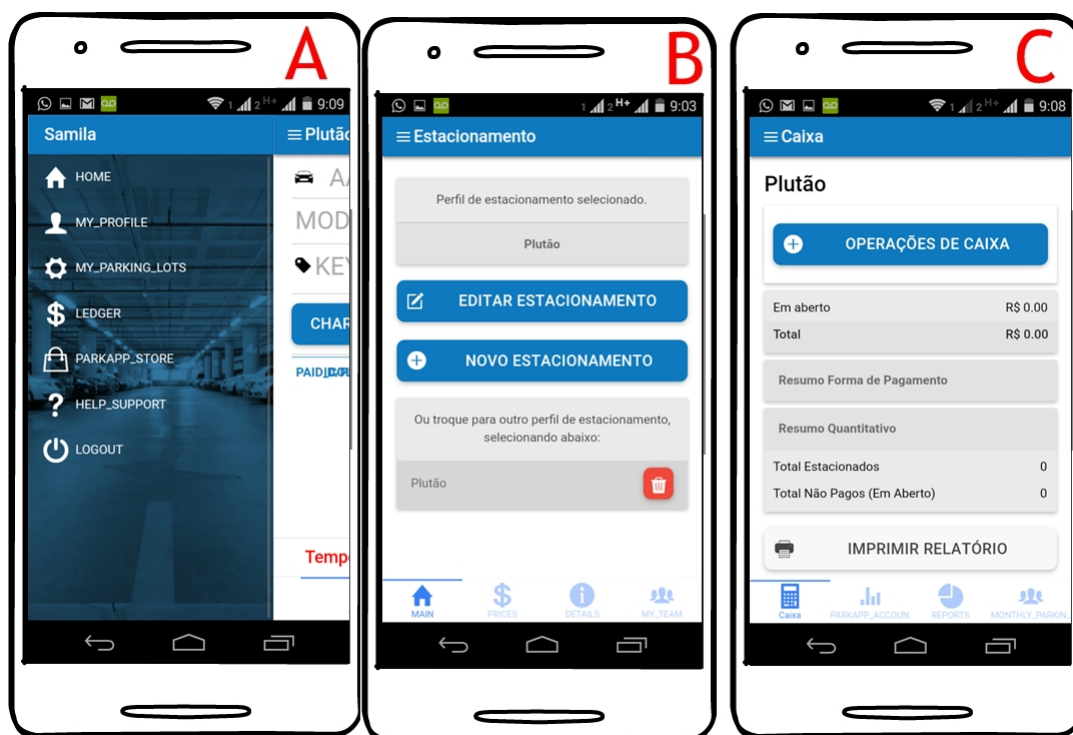
Por meio da palavra *parking* e os filtros para aplicativos gratuitos que possuem quatro

estrelas ou mais e que estão na categoria de Mapas e Navegação uma lista de aplicativos que se enquadravam nestas categorias foram retornados entre eles o ParAli.

3.2.3 ParkApp Admin Estac. e Valets

O aplicativo permite gerenciar mais de um estacionamento, sendo que para cada estacionamento é configurado preços conforme o tipo de estadia, detalhes de pagamento e operação e cadastrar funcionários. Uma vez que o perfil do estacionamento é criado é possível entrar na interface inicial daquele estacionamento e na tela principal podem ser registrados os carros que entram e saem do estacionamento, fazendo o *check-in* e o *check-out*. O *check-in* pode ser personalizado, permitindo ou não tirar foto da placa do carro entre outras opções. O aplicativo também possui uma interface onde é possível fazer o gerenciamento financeiro do estacionamento, a qual permite a geração de relatórios e operações de caixa. O aplicativo (Figura 9) fica disponível gratuitamente por um tempo, depois da expiração deste tempo é necessário pagar por uma assinatura mensal, semestral ou anual.

Figura 9 – Telas do aplicativo ParkApp Admin Estac. e Valets



Fonte: Adaptado de [Internacional \(2016\)](#)

A Figura 9.A apresenta o menu do aplicativo, com as opções disponíveis para o usuário, seguido da tela principal de gerenciamento dos estacionamentos (Figura 9.B), na qual é possível cadastrar um novo estacionamento ou editar um existente e a tela das operações de caixa (Figura

9.C), a qual exibe um relatório das finanças do estabelecimento. Na busca deste aplicativo a *string* de busca apresentada no Quadro 11 foi utilizada.

Quadro 11: String de busca utilizada na pesquisa do ParkApp Admin Estac. e Valets

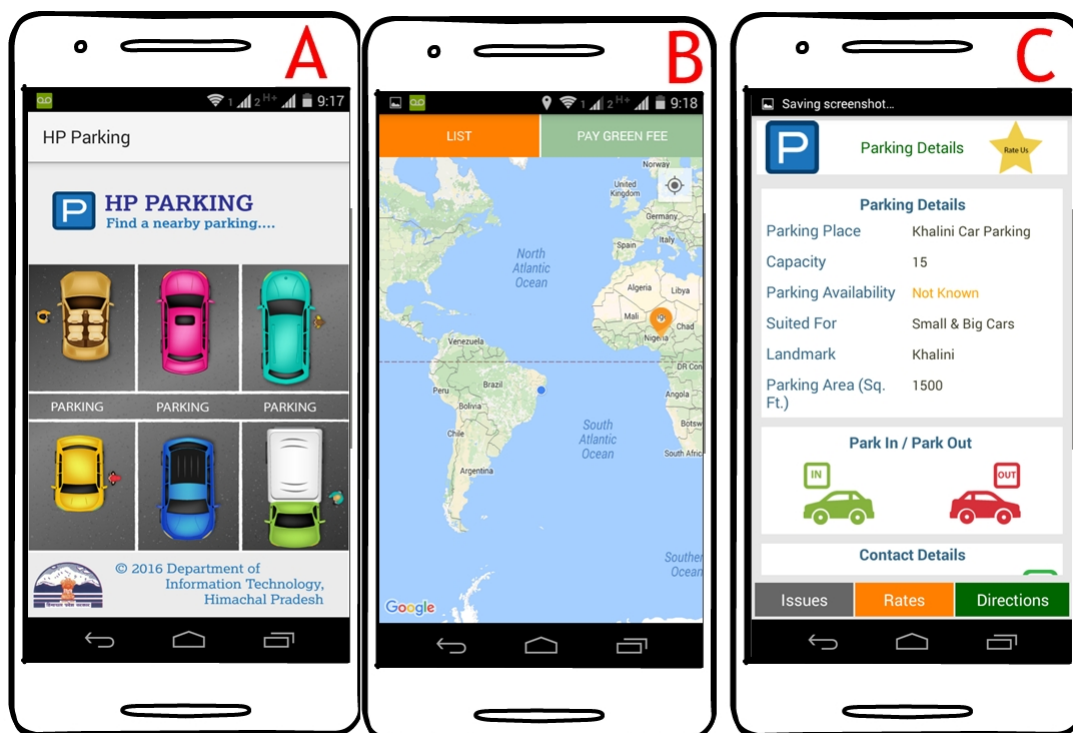
String: *parking manager* + Preço: Gratuitos + Classificação: Quatro Estrelas ou Mais + Categoria: Mapas e Navegação

A partir das palavras *parking manager* e o uso de filtros para aplicativos gratuitos que possuem classificação de quatro estrelas ou mais e se enquadram na categoria Mapas e Navegação uma lista de aplicativos foram retornados entre eles o ParkApp Admin Estac. e Valets.

3.2.4 Himachal Parking

O aplicativo necessita de um cadastro prévio antes que seja possível entrar na interface principal, na qual é apresentado um mapa com a localização atual do usuário, utilizando a opção *list* no menu superior é possível ter acesso a todos os estacionamentos cadastrados, clicando em um dos estacionamentos da lista são mostrados detalhes do estacionamento, com a opção de mostrar o caminho que deve ser tomado para chegar no estacionamento escolhido. Na tela inicial aparece um ícone indicando o estacionamento e ao clicar no ícone informações sobre aquele estacionamento são mostradas na tela, como reputação, capacidade, ponto de referência, tempo mínimo de estacionamento e disponibilidade. A Figura 10 apresenta algumas telas deste aplicativo.

Figura 10 – Telas do aplicativo Himachal Parking



Fonte: Adaptado de Pradesh (2016)

A Figura 10.A mostra a tela inicial do aplicativo, seguido do mapa (Figura 10.B) no qual são exibidas as vagas e por fim, a Figura 10.C exibe a tela de informações do estacionamento selecionado pelo usuário. Na busca por este aplicativo a *string* apresentada no Quadro 12 foi utilizada.

Quadro 12: String de busca utilizada na pesquisa do Himachal Parking

String: *parking manager* + Preço: Grátis + Classificação: Quatro Estrelas ou Mais + Categoria: Turismo e Local

A partir das palavras *parking manager* e o uso de filtros para aplicativos gratuitos que possuem classificação de quatro estrelas ou mais e se enquadram na categoria Turismo e Local, uma lista de aplicativos foram retornados entre eles o Himachal Parking.

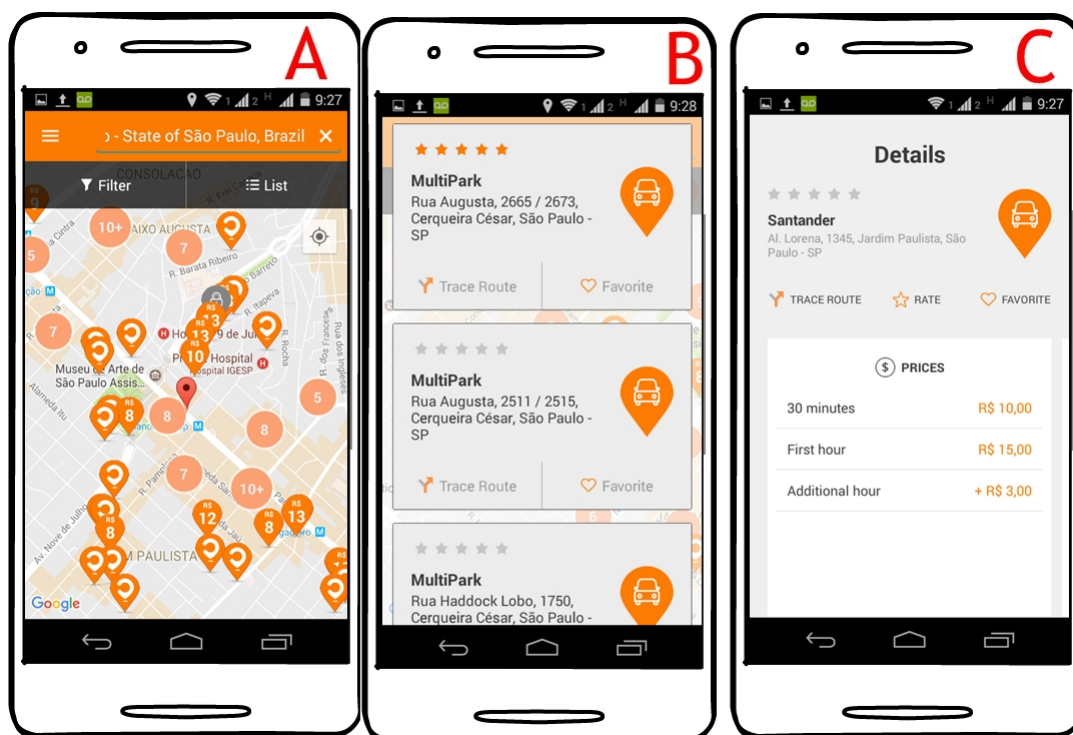
3.2.5 Onde Parar

A tela inicial apresenta um mapa com a localização atual do usuário e os estacionamentos próximos. É possível aplicar filtros do tipo de vaga, métodos de pagamento e filtrar por estacionamentos que oferecem lavagem do carro e os que oferecem descontos. Permite logar no sistema com o Facebook¹⁰ e também é possível criar um cadastro. Quando é realizado o cadastro, é

¹⁰ Facebook: <https://www.facebook.com/>

necessário validar por meio de um código que é enviado por *email* ou *sms*. Os estacionamentos são exibidos no mapa com um ícone indicando a sua localização e ao clicar no ícone detalhes (preços, endereço, avaliação) do estacionamento são exibidos e é possível traçar uma rota para este estacionamento, para isso o aplicativo abre o Google Maps¹¹. Também é possível visualizar todos os estacionamentos próximo disponíveis por meio de uma lista, acessível na opção *list* presente na tela. Algumas telas deste aplicativo são apresentadas na Figura 11.

Figura 11 – Telas do aplicativo Onde Parar



Fonte: Adaptado de OndeParar (2016)

A Figura 11.A exibe a tela inicial do aplicativo, seguido da lista dos estacionamentos (Figura 11.B) cadastrados próximo ao endereço informado e a tela dos detalhes do estacionamento representada na Figura 11.C. Para busca deste aplicativo a *string* apresentada no Quadro 13 foi utilizada.

Quadro 13: String de busca utilizada na pesquisa do OndeParar

String: aplicativos para encontrar estacionamento Categoria: Mapas e Navegação

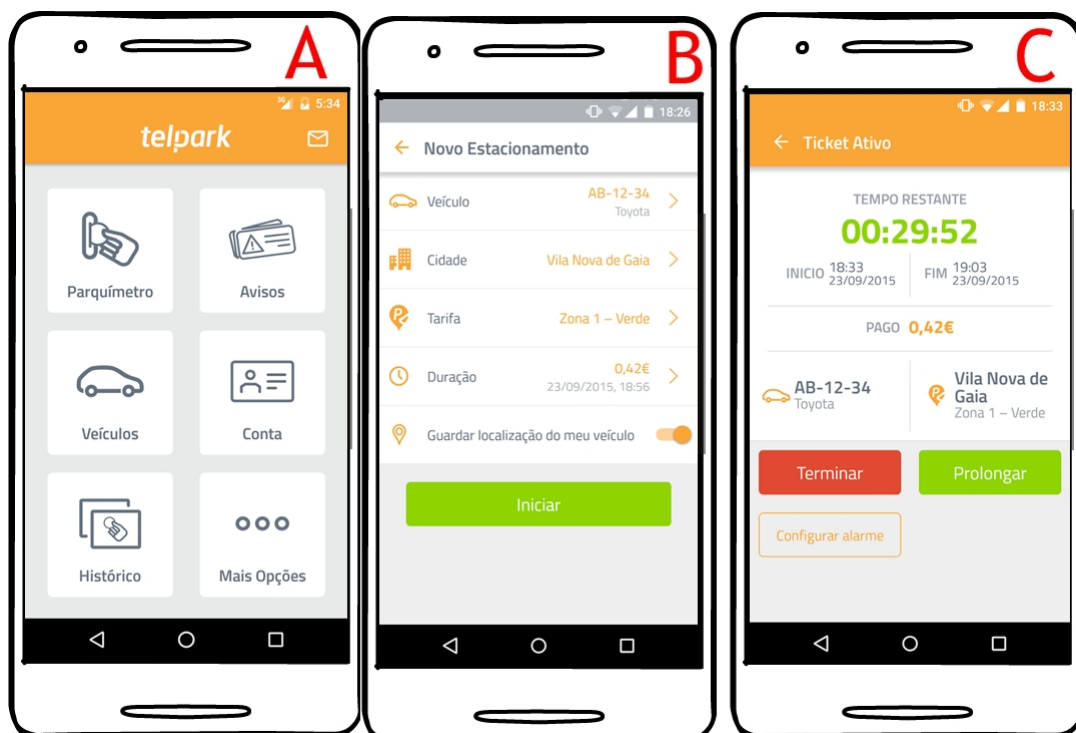
A partir da frase "aplicativos para encontrar estacionamento" e o uso de filtro para categoria Mapas e Navegação, uma lista de aplicativos foi retornada entre estes o Onde Parar.

¹¹ Google Maps: <https://www.google.com.br/maps/>

3.2.6 Telpark

O aplicativo é focado nos estacionamento das vias públicas. Após criar uma conta no serviço, é possível cadastrar os veículos que serão estacionados por meio do *app*, iniciar o parquímetro, receber notificações concernentes ao tempo estacionado e aos valores pagos, consultar o histórico de estacionamento e gerenciar o perfil. Adicionalmente, permite realizar a reserva de vaga nos estacionamentos de aeroportos. Tem como restrições a necessidade de se cadastrar um meio de pagamento antes de realizar qualquer operação e possui um alcance limitado a Portugal e Espanha. A Figura 12 apresenta algumas telas deste aplicativo.

Figura 12 – Telas do aplicativo TelPark



Fonte: Adaptado de [ESPAÑA \(2017\)](#)

A Figura 12.A mostra a tela principal do aplicativo, na qual são exibidas as opções disponíveis neste, seguido da tela de cadastro de novo estacionamento (Figura 12.B) e por fim a Figura 12.C exibe a tela do contador do tempo estacionado. Na busca deste aplicativo a *string* apresentada no Quadro 14 foi utilizada.

Quadro 14: *String* de busca utilizada na pesquisa do Telpark

String: *parking manager* + Preço: Gratuitos + Classificação: Quatro Estrelas ou Mais + Categoria: Mapas e Navegação

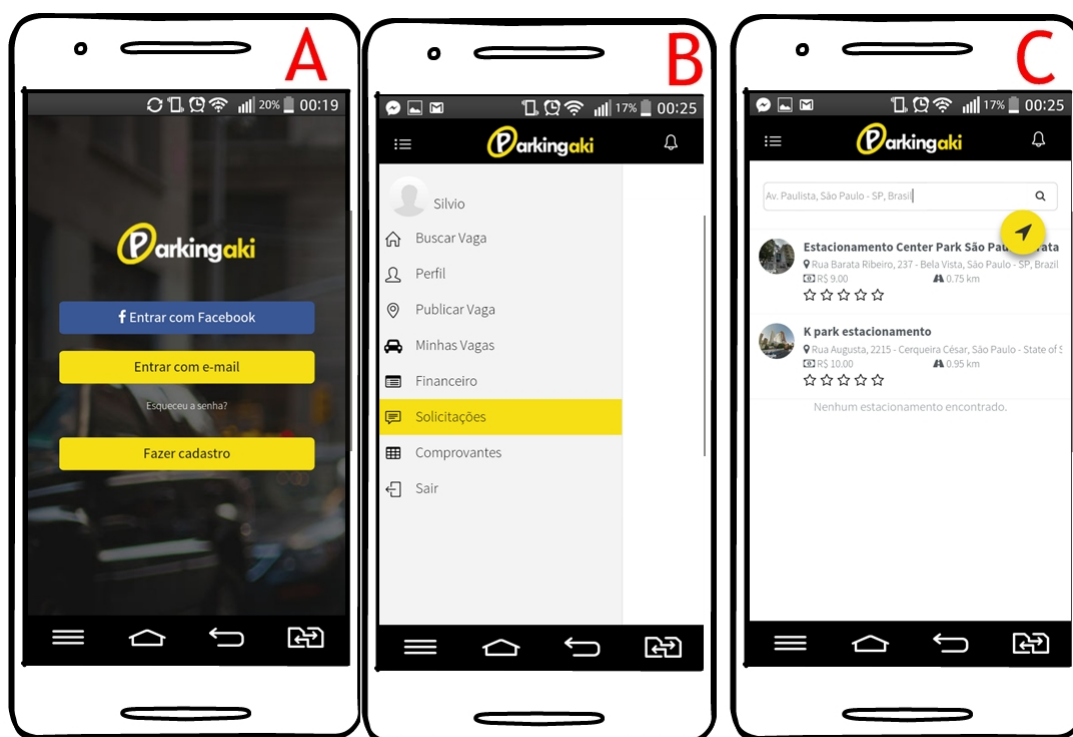
A partir das palavras *parking manager* e utilizando filtros para aplicativos gratuitos que possuem classificação de quatro estrelas ou mais e se enquadram na categoria Mapas e

Navegação uma lista de aplicativos foi retornada entre estes o Telpark.

3.2.7 ParkingAki

A tela inicial pede que o usuário faça o login ou se cadastre. Após o cadastro, imediatamente é possível acessar as funcionalidades do sistema. A tela principal permite que usuário digite um endereço (ou um CEP) e retorna uma lista dos estacionamentos na região, mostrando também seus valores. O menu possui uma série de opções: Buscar vaga, perfil, publicar vaga (onde o usuário cadastra suas próprias vagas de estacionamento, que ficarão visíveis no sistema), minhas vagas (onde o usuário gerencia as vagas que cadastrou), financeiro (onde o usuário vê o quanto arrecadou por meio do aplicativo), solicitações (exibe as solicitações de reserva de vagas), comprovantes e sair. A interface do aplicativo (Figura 13) permite fácil navegação entre suas funcionalidades.

Figura 13 – Telas do aplicativo Parking Aki



Fonte: Adaptado de Taveira (2016)

A Figura 13.A exibe a tela de login do aplicativo, a qual permite fazer login via Facebook, seguido da tela do menu de opções (Figura 13.B), a qual apresenta as opções disponíveis para o usuário e por fim, a Figura 13.C exibe a tela com a lista de estacionamentos cadastrados no sistema que estão localizados próximos ao endereço informado. Para realizar a busca deste aplicativo a *string* apresentada no Quadro 15 foi utilizada.

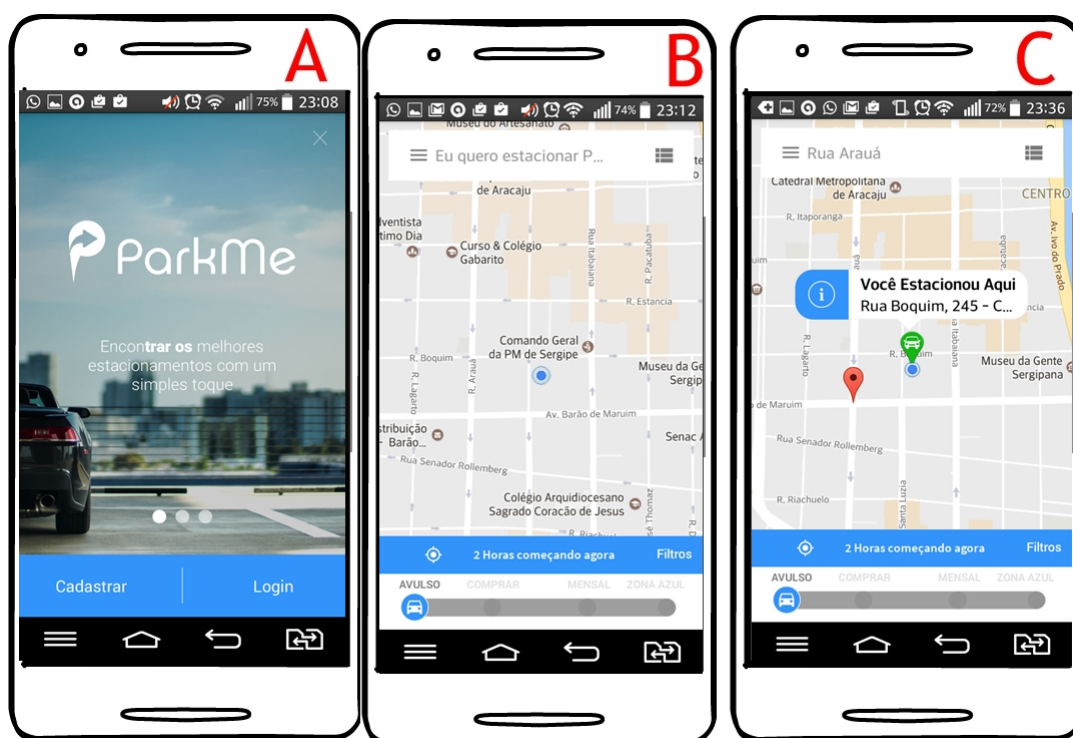
Quadro 15: *String* de busca utilizada na pesquisa do ParkingAki**String:** aplicativos para encontrar estacionamento Categoria: Mapas e Navegação

Utilizando a sentença "aplicativos para encontrar estacionamento" e definindo a categoria como Mapas e Navegação uma lista de aplicativos foi retornada entre estes o ParkingAki.

3.2.8 ParkMe

A tela inicial do aplicativo apresenta as opções de efetuar login ou cadastrar-se. Após concluir o cadastro, o usuário é redirecionado para a tela inicial. Nesta tela há um mapa e um buscador que permite encontrar estacionamentos na região do destino desejado. Os estacionamentos são filtrados de acordo com preço e distância do destino. Ao marcar que estacionou o veículo, o local de estacionamento passa ser exibido com um pino no mapa e é possível verificar-lo a qualquer momento escolhendo a opção "Localizar meu carro" no menu. É possível ainda adicionar um *timer* que enviará uma notificação assim que o tempo marcado para o estacionamento se esgotar. O usuário pode também cadastrar seu veículos e alterar demais informações no cadastro. Há uma opção de adicionar um cartão de crédito para efetuar pagamentos por meio do aplicativo. Algumas telas deste aplicativos são apresentadas na Figura 14.

Figura 14 – Telas do aplicativo ParkMe



Fonte: Adaptado de INRIX (2017)

A Figura 14.A exibe a tela de login do aplicativo, seguida da tela principal (Figura 14.B),

na qual é exibido um mapa com a localização atual do usuário e por fim a Figura 14.C exibe a tela de localizar o carro estacionado, a qual exibe a posição do carro e a posição atual do usuário e traça uma rota entre estes. Este aplicativo foi encontrado a partir da utilização da *string* de busca apresentada no Quadro 16.

Quadro 16: String de busca utilizada na pesquisa do ParkMe

String: aplicativos para encontrar estacionamento Categoria: Mapas e Navegação

A *string* de busca foi formada pela sentença "aplicativos para encontrar estacionamento" e pela definição da categoria Mapas e Navegação. Uma lista de aplicativos foi retornada entre eles estava o ParkMe.

A partir da exploração dos 8 aplicativos 12 características foram extraídas. As principais características encontradas nos aplicativos estão discriminadas no Quadro 17, assim como suas respectivas identificações.

Quadro 17: Principais características identificadas nos aplicativos

Identificação	Característica
C1	Localizar Carro
C2	Pagamento Automático
C3	Notifica tempo
C4	Armazena Histórico
C5	Permite Reserva
C6	Estacionamento automático (via <i>bluetooth</i>)
C7	Tira foto da vaga
C8	Gerenciamento Financeiro (Para Estacionamentos)
C9	Tira Foto da placa do carro (Para Estacionamentos)
C10	Gera Relatórios (Para Estacionamentos)
C11	Permite buscar vagas
C12	Exibe a localização atual

A partir do levantamento destas características foi construído um quadro associando estas a cada um dos aplicativos. O Quadro 18 exibe quais característica são apresentadas por cada aplicativo.

Quadro 18: Relação entre os aplicativos avaliados e as características associadas

Características Aplicativos	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
Parking: Localizar meu carro				X		X	X					X
ParAli Estacionamento			X								X	X
ParkApp Admin Estac. e Valets								X	X	X		
Himachal Parking											X	X
Onde Parar											X	X
TelPark			X	X	X						X	
ParkingAki								X			X	X
ParkMe	X	X	X								X	X
Smart Parking	X	X	X	X	X	X	X	X	X	X	X	X

Com base nesta análise percebe-se que existem características que são comuns a quase todos os aplicativos, como a capacidade de buscar vagas, e outras específicas a um determinado aplicativo, mas observa-se que nenhum deles oferece os benefícios que todas estas características juntas proporciona. É neste ponto que entra o *Smart Parking*, a solução desenvolvida neste trabalho, que foi elaborada de forma que suprisse as necessidades tanto do motorista que utiliza a ferramenta como do gestor que cadastra seu estacionamento no aplicativo.

3.3 Questionário com possíveis utilizadores para levantamento de requisitos

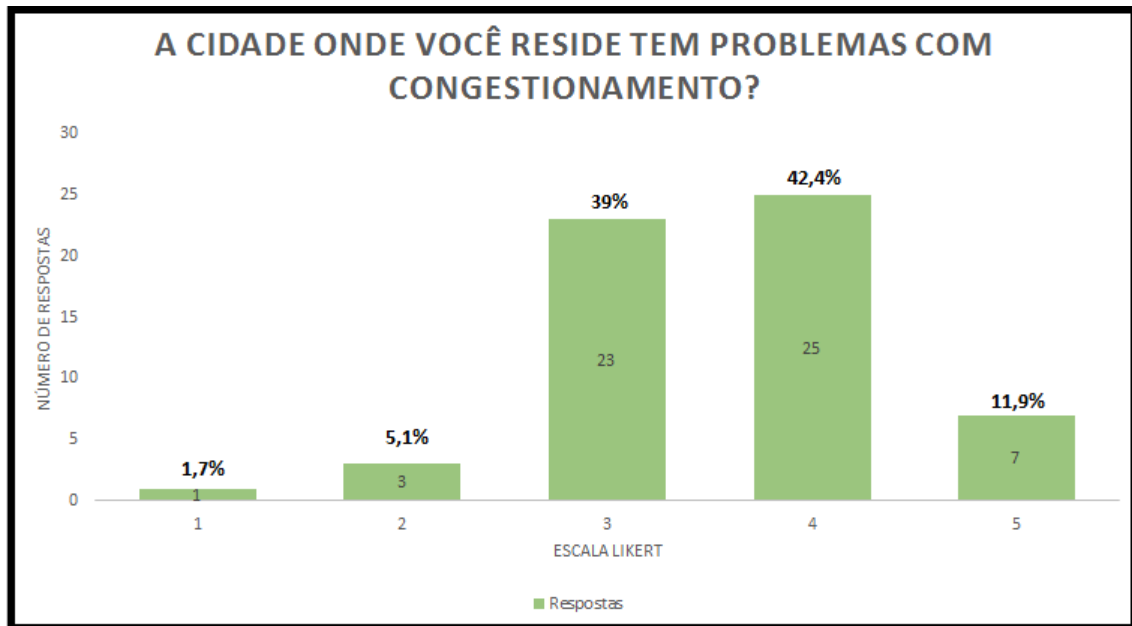
Um questionário, cujas perguntas e respostas obtidas podem ser visualizados no Apêndice A, foi distribuído por meio do Facebook e do WhatsApp¹² para a coleta da opinião dos respondentes sobre a situação no que diz respeito a estacionamentos na cidade em que este reside. 59 pessoas responderam o questionário que foi dividido em 9 questões.

A primeira pergunta teve como objetivo descobrir o contexto da cidade em que os respondentes vivem e foi questionado se a cidade onde residem possui problemas de congestionamento. Para esta análise, na maioria das perguntas, foi utilizada a escala *Likert* (1932), um tipo de escala que considera o nível de concordância que os respondentes tem com a pergunta em questão e varia do nível 0, não concordo, a nível 5, concordo totalmente.

Para esta pergunta, 54,3% das pessoas responderam que a cidade onde habitam possuem problemas de estacionamento em um nível alto ou muito alto, o que significa grande dificuldade para encontrar locais para estacionar em horários de pico, congestionamentos e pouca fluidez no tráfego de veículos. As resposta estão demonstradas na Figura 15.

¹² WhatsApp: <https://www.whatsapp.com/>

Figura 15 – Gráfico das respostas da pergunta 1 do questionário para levantamento de requisitos do Smart Parking.

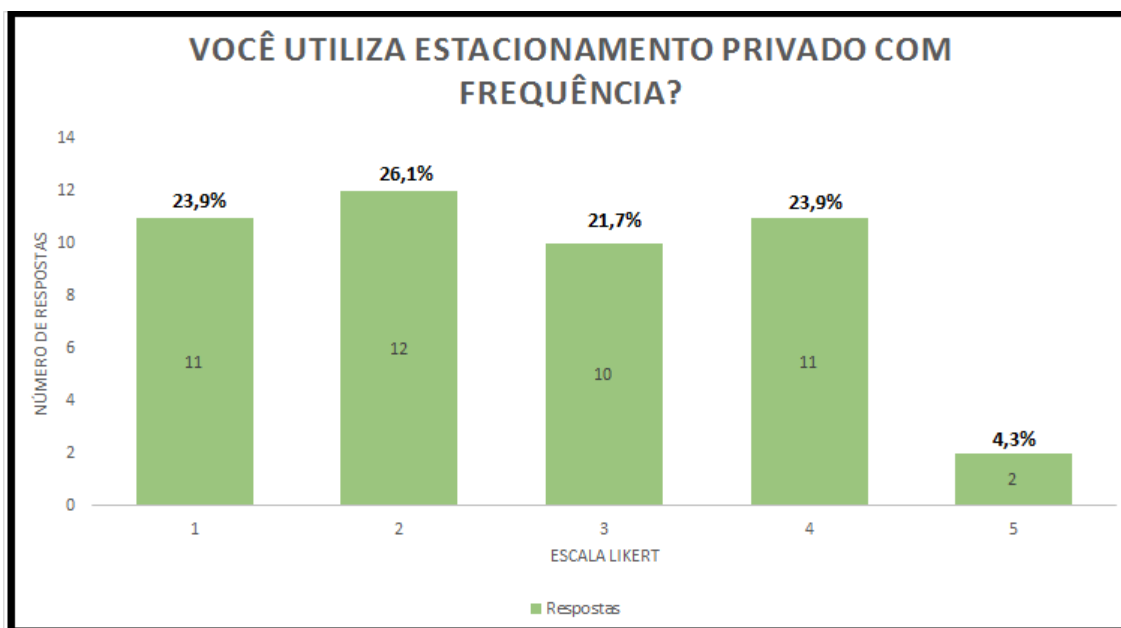


Fonte: Próprios autores

Na segunda questão foi abordado quanto tempo os respondentes gastam na procura por estacionamento e 88,1% deles informaram que gastam tempo médio na busca de uma vaga contra 5,1% que respondeu que leva muito tempo e 6,8% informou que gasta pouco tempo na procura de uma vaga para estacionar.

Quando abordado sobre o uso de estacionamentos privados, pouco mais de metade dos entrevistados responderam usar estacionamentos privados, 59,3% destes.

Figura 16 – Gráfico das respostas da pergunta 4 do questionário para levantamento de requisitos do Smart Parking.



Fonte: Próprios autores

Para os que responderam que utilizam estacionamentos privados, notou-se que a frequência com que usa é de baixa a moderada (Figura 16).

Também foi questionado se os respondentes tem dificuldades em lembrar onde estacionou o carro e das 59 respostas, apenas 2 responderam que esquecem onde estacionou com frequência.

Foi perguntado sobre os aplicativos mais conhecidos que auxiliam os motoristas a encontrar uma vaga para estacionar (ParAliEstacionamento¹³, OndeParar¹⁴, ParkMe¹⁵, ParkingAki¹⁶, Google Open Spot, Waze¹⁷), qual que o respondente já utilizou e nota-se que poucas pessoas utilizam aplicativos para este fim, apenas 3 destas responderam, como mostrado na Figura 17.

¹³ Para baixar o aplicativo ParAli Estacionamento acesse <https://goo.gl/xngSRk>

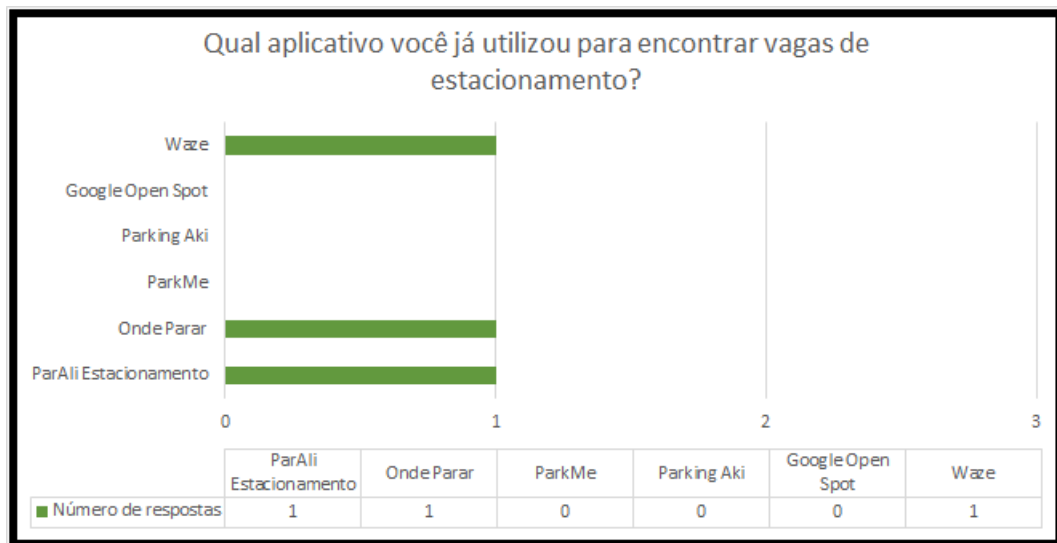
¹⁴ Para baixar o aplicativo OndeParar acesse <https://goo.gl/KbHI7f>

¹⁵ Para baixar o aplicativo ParkMe acesse <https://goo.gl/5JvdR5>

¹⁶ Para baixar o aplicativo ParkingAki acesse <https://goo.gl/QRDvG6>

¹⁷ Para baixar o aplicativo Waze acesse <https://goo.gl/L9woxS>

Figura 17 – Gráfico das respostas da pergunta 7 relativo aos aplicativos utilizados na procura por vagas de estacionamento do questionário para levantamento de requisitos do Smart Parking.

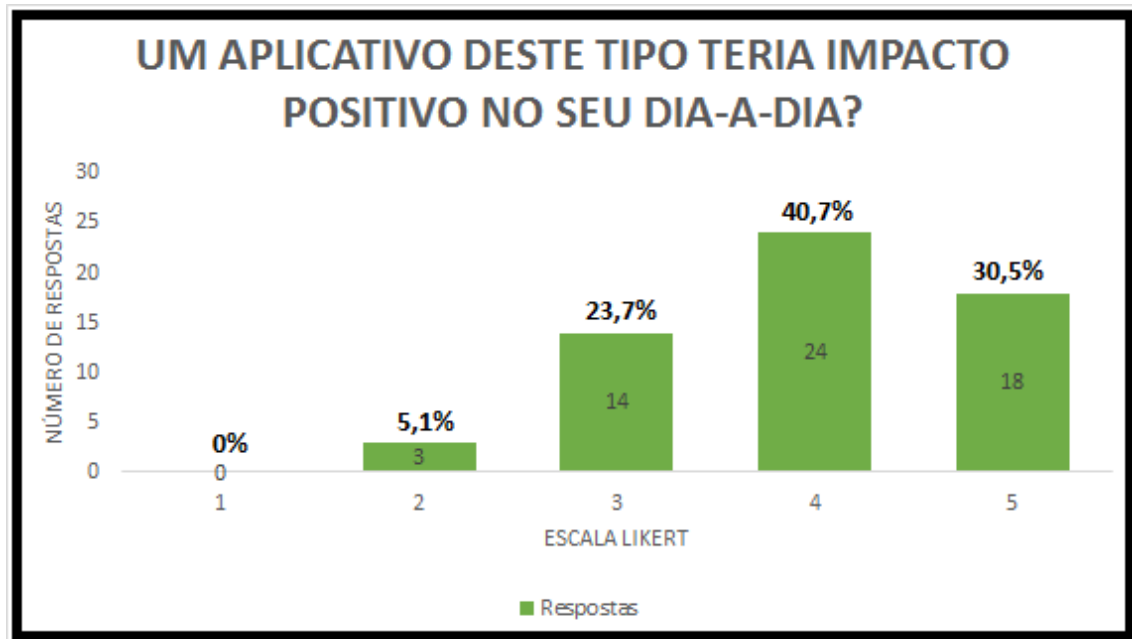


Fonte: Próprios autores

Dos seis aplicativos apresentados na pergunta, apenas três foram utilizados pelos três respondentes, um para cada aplicativo.

Por fim, quando questionado o impacto que um aplicativo como este teria na rotina dos entrevistados 40,7% responderam que teria um impacto significativo, como mostrado na Figura 18.

Figura 18 – Gráfico das respostas da pergunta 9 do questionário para levantamento de requisitos do Smart Parking.



Fonte: Próprios autores

Observa-se que um alto número de pessoas utilizam estacionamento privado e embora poucas pessoas já utilizem aplicativos no auxílio da busca por estacionamentos com vagas disponíveis, os entrevistados apontaram fatores que os fariam utilizar uma ferramenta deste tipo como a praticidade proporcionada e a otimização do tempo nesta busca. De acordo com as respostas, nota-se que uma vez que a ferramenta seja fácil de utilizar, tenha uma boa divulgação e apresente características que trazem vantagem para o usuário, como notificação do tempo que o veículo está estacionado, a localização do carro através do mapa e a possibilidade de fazer a reserva de uma vaga tem um impacto significativo no dia-a-dia destes usuários.

4

Desenvolvimento do Produto de Software

Esse capítulo apresenta o projeto do *Smart Parking*, descrevendo os Requisitos Funcionais e Não-Funcionais, além do diagrama de Casos de Uso, o Protótipo de Telas, Diagrama de Classe, Diagrama de Pacote, Planejamento das *Sprints* e o Esquema do Banco de Dados.

4.1 Ferramentas Utilizadas

O Android Studio¹ é a IDE (Figura 19) oficial para desenvolvimento de aplicações para a plataforma Android. Ele foi anunciado pela Google em Maio de 2013, na Google I/O Conference². Sua primeira versão estável foi disponibilizada em Dezembro de 2014. Ele foi baseado no software IntelliJ IDEA³ da JetBrains⁴ e está disponível para download em versões para os Sistemas Operacionais Windows, Mac OS e Linux (DUCROHET; NORBYE; CHOU, 2013). E como as aplicações Android são escritas em Java, é preciso instalar também uma versão atualizada do JDK⁵ (Java Development Kit) para desenvolver aplicações com o Android Studio.

¹ Android Studio: <https://developer.android.com/studio/index.html?hl=pt-br>

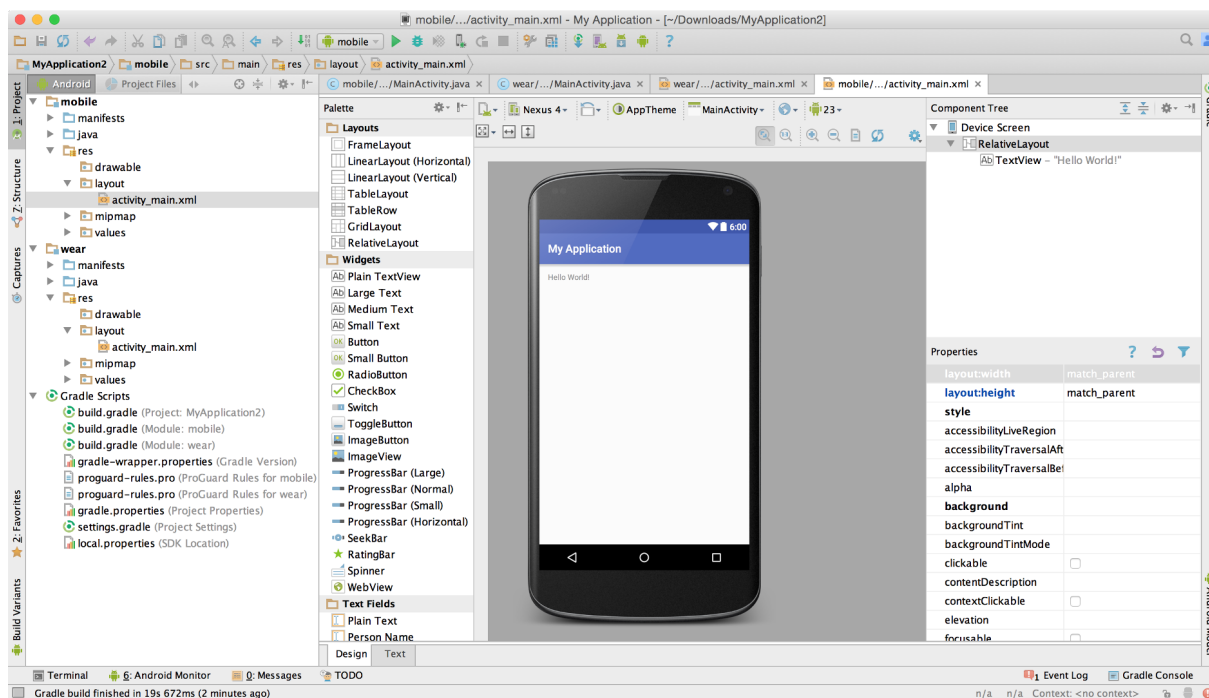
² As palestras da conferência estão disponibilizadas em sua página oficial: <https://developers.google.com/events/io/2013/>

³ IntelliJ IDEA: <https://www.jetbrains.com/idea/>

⁴ JetBrains: <https://www.jetbrains.com/>

⁵ Disponível para download no site da Oracle: <http://www.oracle.com/technetwork/pt/java/javase/downloads/jdk8-downloads-2133151.html>

Figura 19 – Ambiente de desenvolvimento - Android Studio



Fonte: Android Studio

Como foi criado especificamente para o desenvolvimento de aplicações Android ele possuiu uma série de funcionalidades que facilitam esse desenvolvimento, como as seguintes: *instante run* (ele compreende as alterações de forma inteligente e as distribui sem reiniciar o aplicativo nem recompilar o APK, permitindo a visualização imediata dos efeitos das alterações), editor de código inteligente, Android Emulator (instala e inicializa aplicativos mais depressa que os dispositivos reais, facilitando os testes durante o desenvolvimento), sistema de compilação robusto, integração com ferramentas de controle de versão (GIT⁶, Mercurial⁷ e Subversion⁸), ferramentas de teste, editor de layout, APK Analyzer (permite inspecionar o conteúdo do aplicativo em desenvolvimento) (GOOGLE, 2017).

O Android Studio foi utilizado no desenvolvimento da aplicação proposta, mas antes da implementação do software foi realizado todo o planejamento necessário e para isso utilizou-se a plataforma Trello⁹ para auxiliar na organização e delineamento do projeto.

O Trello é uma ferramenta flexível utilizada para o gerenciamento de projetos. É gratuito e foi desenvolvido pela *Fog Creek Software*¹⁰ em 2011. Trata-se de uma ferramenta de quadros virtuais que pode ser usado para o gerenciamento de tarefas, principalmente àquelas voltadas ao desenvolvimento de software com Scrum.

⁶ Git: <https://git-scm.com/>

⁷ Mercurial: <https://www.mercurial-scm.org/>

⁸ Subversion: <https://subversion.apache.org/>

⁹ Trello: <https://trello.com/>

¹⁰ Fog Creek Software: <http://www.fogcreek.com/>

Ele permite utilizar o paradigma Kanban para gerenciar projetos. O Kanban foi originalmente popularizado pela Toyota nos anos 80 para gerenciamento da cadeia de suprimentos. No Trello, os projetos são representados por *boards*(quadros), que contêm listas (correspondentes a listas de tarefas). As listas contêm cartões (correspondentes a atividades).

Esses cartões podem ser movimentados entre as colunas de um mesmo quadro, assim é possível realizar transições de um tópico para as demais colunas simplesmente arrastando o cartão. Os cartões são como notas, contudo eles podem receber informações complexas, sendo possível adicionar prazos, *checklists* e anexar arquivos a um cartão. Assim é possível verificar o progresso das atividades em níveis de porcentagem. Além disso, os usuários podem ser atribuídos a cartões, o que ajuda a gerenciar projetos em equipe. Usuários e *boards* (projetos) também podem ser agrupados em organizações (CASTELLI, 2015).

4.2 Configuração da máquina de desenvolvimento

Para o desenvolvimento desse estudo foram utilizados dois notebooks: um Lenovo¹¹ IdeaPad G400s (Quadro 19) e um Dell¹² Inspiron 3420 (Quadro 20).

Quadro 19: Configurações do notebook Lenovo

Produto	Marca	Especificação
Processador	Intel	Intel Core i5-3230M 2.6 GHz
Disco rígido	-	1 Tb
Memória RAM	-	4Gb
Dispositivo gráfico	Intel	Intel HD Graphics 4000
Sistema Operacional	Microsoft	Windows 10

Quadro 20: Configurações do notebook Dell

Produto	Marca	Especificação
Processador	Intel	Intel Core(TM) i3-2328M 2.2 GHz
Disco rígido	-	650Gb
Memória RAM	-	4Gb
Dispositivo gráfico	Intel	Intel HD Graphics 3000
Sistema Operacional	Linux	Ubuntu 16.4 LTS

4.3 Requisitos de Software

Pressman (2011, p.126) acredita que entender o que o cliente deseja para o sistema, ou seja, "os requisitos do sistema está entre as tarefas mais difíceis que um engenheiro de software pode enfrentar", isso não é por acaso para Sommerville (2007, p.81) esses requisitos "dependem

¹¹ Lenovo: <http://www.lenovo.com/br/pt/>

¹² Dell: <http://www.dell.com.br/>

do tipo de software que está sendo desenvolvido, dos usuários a que o software se destina e da abordagem geral considerada pela organização ao redigir os requisitos", todos estes detalhes devem ser considerados na definição das funcionalidades diretas e indiretas do sistema e devido a subjetividade do processo a tarefa se torna complexa.

4.3.1 Requisitos Funcionais

Os Requisitos Funcionais do sistema são referentes ao que o sistema deve fazer, ou seja quais são suas funções principais. O Quadro 21 especifica e descreve os Requisitos Funcionais do produto de software proposto.

Quadro 21: Requisitos Funcionais

	Requisito	Prioridade	Descrição
RF01	Manter Usuários	Essencial	Capacidade de cadastrar, editar e excluir usuários que acessam o sistema.
RF02	Manter Estacionamentos	Essencial	Capacidade de cadastrar, editar e excluir estacionamentos no sistema. Além de permitir funções de check-in e check-out durante o período de funcionamento do estacionamento em questão.
RF03	Manter Histórico de usuário	Essencial	Manter histórico de utilização anteriores do motorista a diferentes estacionamentos, isso pode auxiliar a apresentar sugestões futuras para este usuário.
RF04	Gerenciar Pagamento	Essencial	Realizar cadastro de cartão e permitir comunicação com a prestadora do cartão de crédito para débitos automático.
RF05	Manter vagas	Essencial	Adicionar e excluir vagas de estacionamento além de alterar o status (ocupado, reservado ou disponível) das vagas de estacionamento.
RF06	Consultar Faturamento	Essencial	Listar o faturamento (mensal ou anual) do estacionamento baseado nos dados coletados durante o período.
RF07	Enviar Notificações	Essencial	Enviar notificações quando o tempo de reserva da vaga estiver esgotando e quando o tempo de estadia estiver acabando.
RF08	Manter Reservas de Vagas	Essencial	Cadastrar ou excluir reserva de uma vaga de estacionamento.
RF09	Localizar carro estacionado	Essencial	Localizar onde o carro foi estacionado e traçar uma rota do local onde o motorista se encontra até o local onde o carro foi estacionado.
RF10	Manter Veículo	Essencial	Capacidade de cadastrar, editar e excluir veículos utilizados pelo usuário no sistema.
RF11	Buscar Vagas	Essencial	Capacidade de localizar e exibir vagas disponíveis próximas a um endereço de destino informado pelo usuário.

4.3.2 Requisitos Não-Funcionais

Também conhecidos como Requisitos de Qualidade, os Requisitos Não-Funcionais referem-se ao comportamento e a estrutura do produto de software, sua capacidade de ser usável, seguro, confiável, eficiente e de fácil manutenção. O Quadro 22 especifica e descreve os Requisitos Não-Funcionais associados ao produto de software proposto.

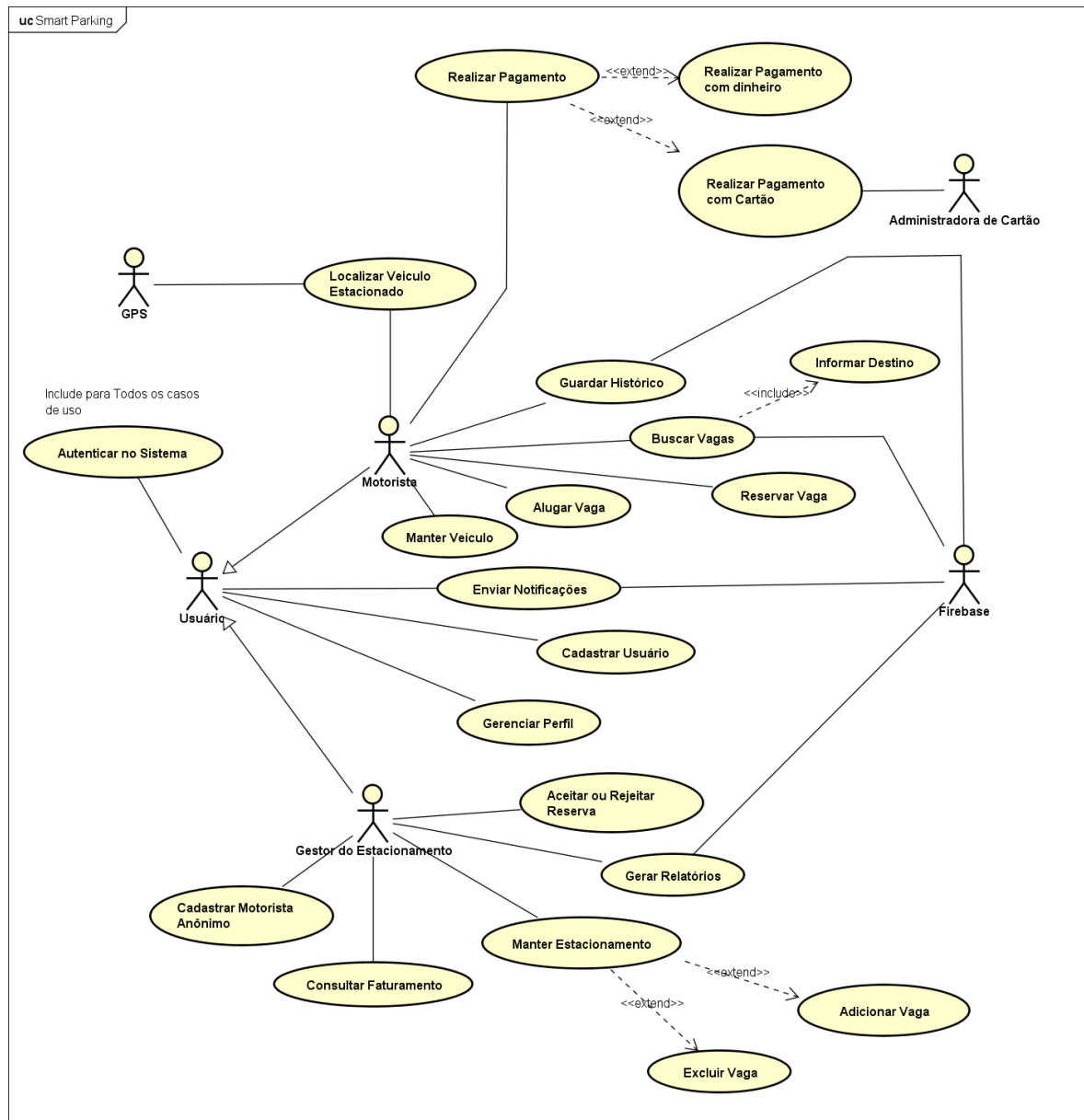
Quadro 22: Requisitos Não-Funcionais

	Requisito	Prioridade	Descrição
RNF01	Baixo Consumo de energia	Essencial	O aplicativo deve trabalhar consumindo o mínimo de energia possível.
RNF02	Usabilidade	Essencial	A interface com o usuário deve apresentar ícones intuitivos, no qual o usuário possa associar cada opção a função que esta é responsável com facilidade.
RNF03	Capacidade para multiusuários	Essencial	Deve suportar vários usuários simultaneamente.
RNF04	Sigilo das informações	Essencial	As informações dos motoristas e gerentes de estacionamento devem ser criptografadas.
RNF05	Controle de Acesso	Essencial	O usuário só está apto a acessar o sistema por meio de autenticação.
RNF06	Linguagem de Desenvolvimento Java	Essencial	A linguagem utilizada no desenvolvimento do software deve ser Java.
RNF07	Plataforma móvel	Essencial	A plataforma alvo no desenvolvimento deve ser a plataforma <i>mobile</i> Android.
RNF08	Sistema de Gerenciamento de Banco de Dados	Essencial	O Banco de Dados utilizado na persistência dos dados é o <i>Realtime database</i> do Firebase.
RNF09	Padrão MVP para a Arquitetura	Essencial	O projeto da arquitetura deve seguir as recomendações do padrão MVP.
RNF10	Consistência dos dados	Essencial	Os dados não podem ser corrompidos ou excluídos por falha do sistema.

4.4 Diagrama de Casos de Uso

Os Casos de Uso enfatizam cenários de como uma aplicação será utilizada e descreve o comportamento de um sistema em diferentes situações de acordo com as entradas dos usuários, estes últimos chamados de Atores, que interagem com o sistema. A Figura 20 apresenta os Atores, os Casos de Uso e as interações referente ao produto de software desenvolvido.

Figura 20 – Diagrama de Casos de Uso - Smart Parking



powered by Astah

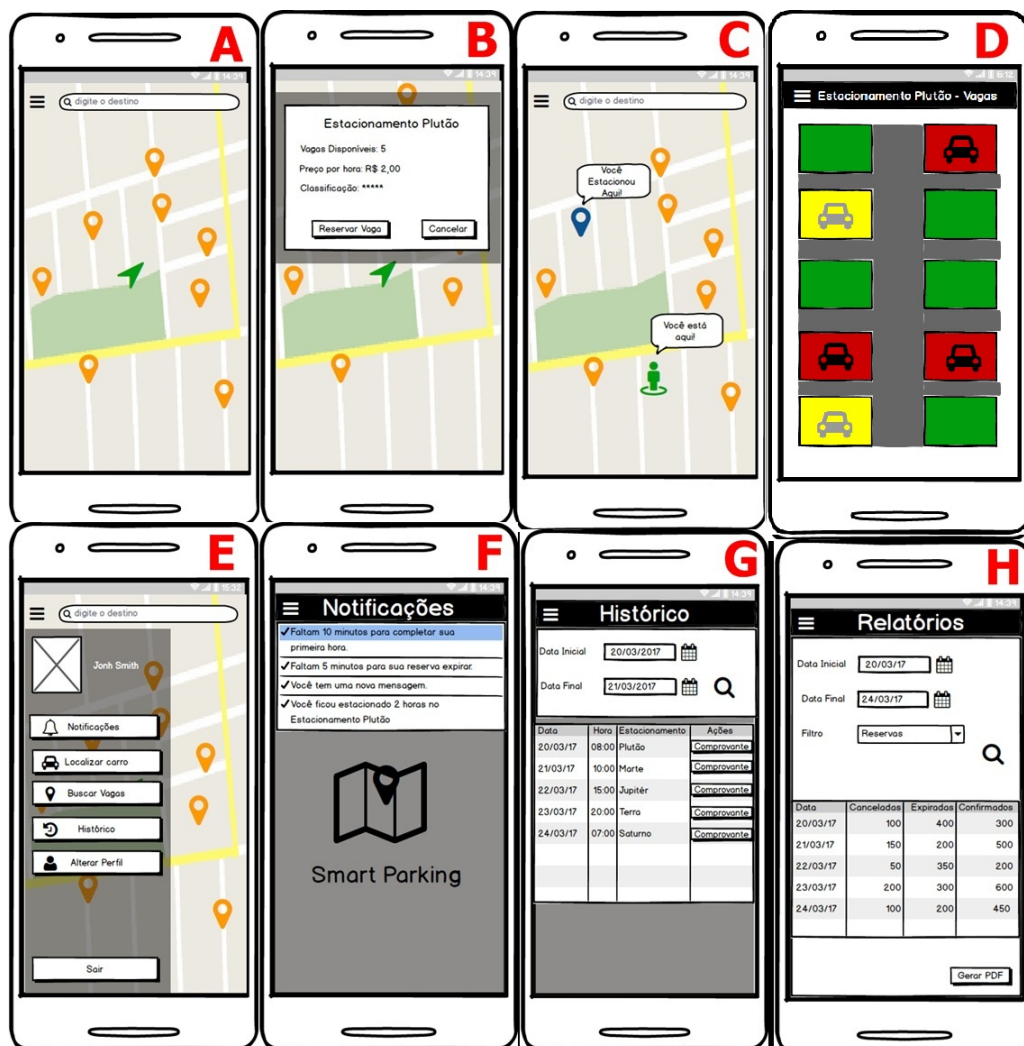
Fonte: Próprios autores

4.5 Protótipo de Telas

Para Wasserman (2010), a experiência do usuário é especialmente crítica, logo há uma grande necessidade de criar protótipo de interfaces do usuário, sendo assim uma vez levantados todos os requisitos necessários para cumprir o objetivo do produto de software elaborado, o protótipo de telas (Figura 21) é o passo seguinte na especificação do projeto em questão ¹³.

¹³ Protótipo clicável de telas em <https://marvelapp.com/544j334>

Figura 21 – Principais telas do protótipo de telas do Smart Parking



Fonte: Próprios autores

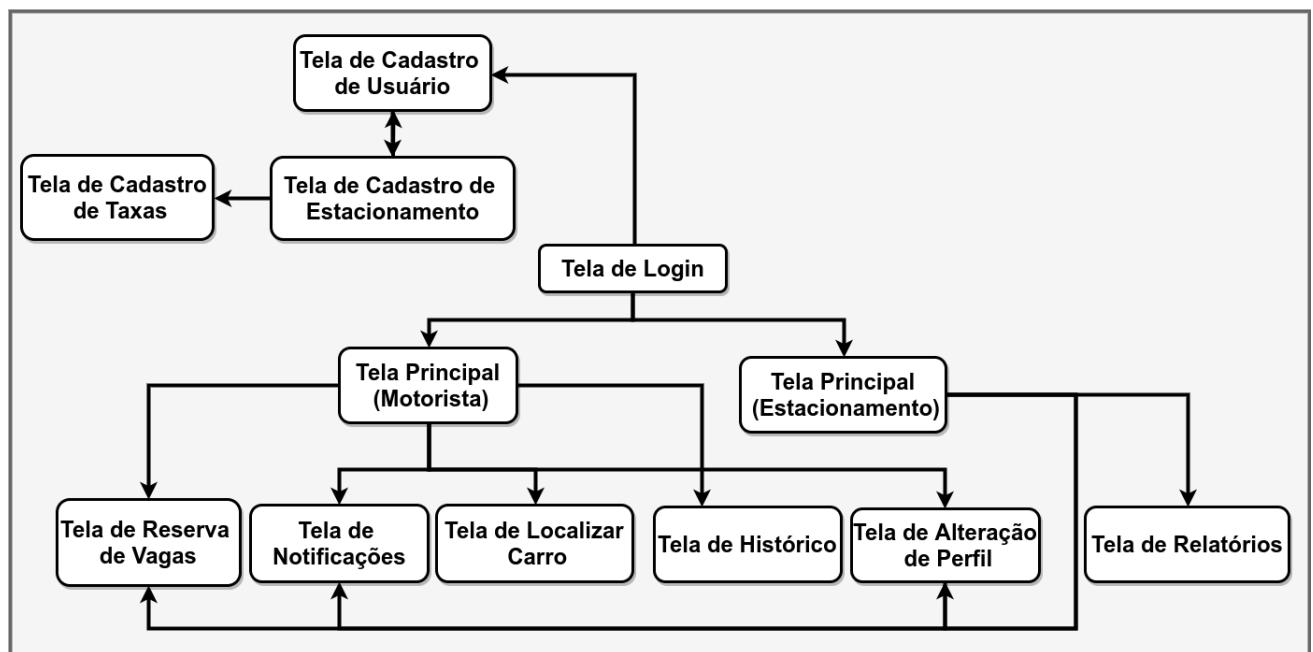
A Figura 21.A exibe a tela inicial do protótipo, a partir da sua localização atual (representada pela seta) o usuário visualiza um mapa da região e por meio do buscador na parte superior da tela pode procurar por um destino, aonde deseja estacionar seu carro. A Figura 21.B apresenta uma janela modal que é exibida quando o usuário seleciona um estacionamento (representados pelos pontos indicativos na tela), nela é possível ver quantas vagas estão livres, o valor cobrado e a classificação do estacionamento. A Figura 21.C apresenta a tela utilizada para localizar o automóvel após o usuário estacionar seu veículo, o local onde ele estacionou fica destacado. A Figura 21.D apresenta o status de ocupação de um estacionamento, sendo que a cor verde representa que a vaga está disponível, vermelha, a vaga está ocupada e amarela, representa que aquela vaga está reservada. A Figura 21.E exibe as opções disponíveis no menu para o usuário. A Figura 21.F exibe a tela de notificações do usuário. A Figura 21.G mostra o histórico de estacionamento do usuário, na qual a partir de uma data inicial e final fornecidas por este são retornados todos os estacionamentos utilizados pelo usuário durante o período. A Figura 21.H

apresenta a tela de relatórios, tela específica para o usuário do tipo Gestor de Estacionamento, a qual retorna, a partir de uma data inicial, uma data final e a definição de filtros, dados relativos ao número de reservas e a utilização do estacionamento naquele período.

4.5.1 Hierarquia de Telas do projeto

Baseado no Protótipo de Telas foi criado a hierarquia de telas apresentada na Figura 22. Nesta são descritas as principais telas do aplicativo e o fluxo das interações de cada tela.

Figura 22 – Diagrama de Hierarquia de Telas - Smart Parking



Fonte: Próprios autores

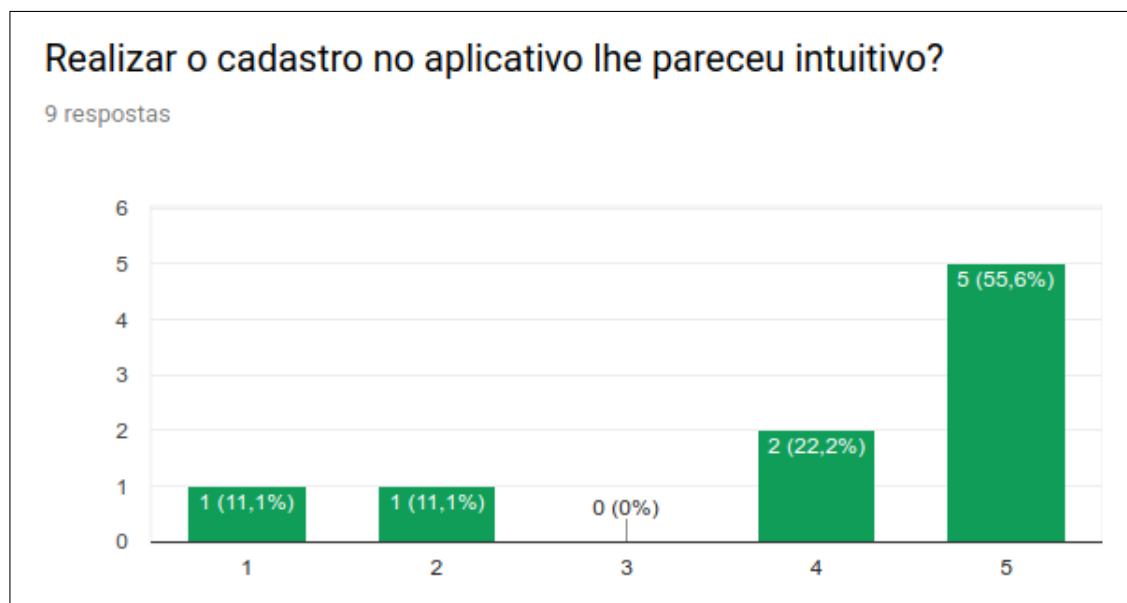
A primeira tela do aplicativo é a Tela de Login, nesta é possível acessar a Tela de Cadastro de Usuário e a Tela de Cadastro de Estacionamento, os usuários podem ser do tipo motorista ou estacionamento (caso estacionamento, é possível acessar a Tela de Cadastro de Taxas do estacionamento). Na Tela de Login também podem ser inseridas as credenciais para acessar o sistema. Quando realizada a autenticação, a primeira página visualizada é a Tela Principal, a qual exibe um mapa com a localização do motorista (quando o usuário é um motorista), ou uma tela com as informações do estacionamento (quando o usuário é do tipo estacionamento), por meio da Tela Principal é possível acessar um menu de opções do aplicativo, o qual permite o acesso a Tela de Notificações, a Tela de Localizar Carro, a Tela de Histórico, a Tela de Reserva de Vaga e a Tela de Alteração do Perfil (caso o usuário seja do tipo motorista) e a Tela de Relatório, Tela de Notificações, Tela de Reserva de Vagas e Tela de Alteração do Perfil (caso o usuário seja do tipo Gestor de Estacionamento).

4.5.2 Validação do protótipo de telas

Para validar o protótipo de telas elaborado, dois questionários foram distribuídos por meio do Facebook no período de 13 de Julho de 2017 até 31 de Julho de 2017, sendo um dos questionário referente ao protótipo de telas direcionado ao Motorista (suas 32 questões e respectivas respostas estão no Apêndice C) e um outro referente ao protótipo de telas direcionado ao Gestor de Estacionamento (suas 28 questões e respectivas respostas estão no Apêndice B). Nestes questionários foram abordadas questões de usabilidade do protótipo, assim como também foram coletadas sugestões dos respondentes em relação a cada tela com a qual o mesmo interagiu. Foram obtidas 9 respostas no questionário que se referia ao protótipo de telas do gestor de estacionamento e 7 respostas no questionário do motorista.

Com base nas respostas coletadas percebe-se que alguns dos respondentes do questionário do Gestor do Estacionamento acharam a tela de cadastro e a definição dos usuários um pouco confusa e as respostas nesta seção variaram mais que as demais.

Figura 23 – Questionário de validação do protótipo de Telas - Tela de Cadastro



Fonte: Google Forms

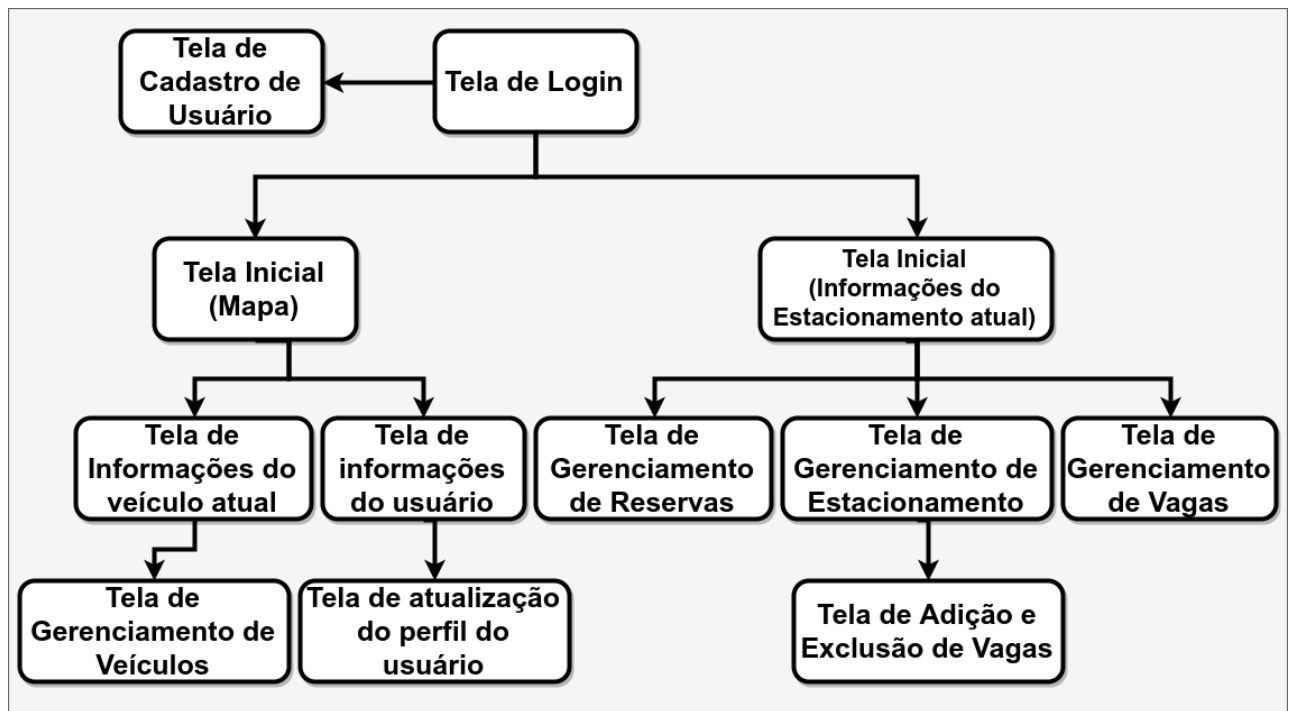
No restante do questionário do Gestor do Estacionamento, de forma geral, os respondentes avaliaram o protótipo de médio a alto nível no que se refere a ser intuitivo e fácil de usar. No questionário do motorista, por sua vez, as respostas também variaram entre notas médias a altas no que se refere a usabilidade das telas do aplicativo.

A partir das respostas conclui-se que de forma geral o protótipo apresenta suas funções de forma intuitiva e apresenta facilidade no seu uso.

4.5.3 Hierarquia de Telas do Produto de Software

Com a experiência e conhecimento adquirido durante a implementação algumas alterações do protótipo original foram realizadas e o protótipo de telas evoluiu para o demonstrado na Figura 24.

Figura 24 – Hierarquia de Telas do protótipo desenvolvido



Fonte: Próprios autores

Nesta hierarquia a partir da Tela de Login é possível navegar para a Tela de Cadastro ou Tela Inicial do respectivo tipo de usuário (Motorista ou Gestor de Estacionamento). Uma vez na Tela Inicial do motorista, na qual é apresentado um mapa, é possível navegar pela Tela de Informações do Veículo atual e a partir desta acessar a Tela de Gerenciamento de Veículo. A partir da tela inicial do motorista também é possível acessar a Tela de Informações do Usuário e a partir desta ir para a Tela de Atualização do Perfil do Usuário.

Uma vez na Tela Inicial do Gestor de Estacionamento, na qual são apresentadas as informações do estacionamento atual, é possível navegar para a Tela de Gerenciamento de Estacionamentos e a partir desta é possível acessar a Tela de Adição e Exclusão de Vagas. Também por meio da Tela inicial do estacionamento é possível acessar a Tela de Gerenciamento de Reservas, Tela de Gerenciamento de Vagas e a Tela de Informações do Usuário e a partir desta última ir para a Tela de Atualização do Perfil do usuário.

4.6 Arquitetura

Para [Ali \(2017\)](#), a arquitetura deve ser o primeiro componente que se deve trabalhar em um projeto de software, pois uma arquitetura bem elaborada minimizará muito retrabalho no futuro de um projeto, assim como provê fácil escalabilidade.

Para o desenvolvimento dessa aplicação foi escolhida a Arquitetura em Camadas. A razão para isso é que essa arquitetura fornece um modelo pelo qual os desenvolvedores podem criar aplicações flexíveis e reutilizáveis. Ao segregar um aplicativo em níveis, os desenvolvedores adquirem a opção de modificar ou adicionar uma camada específica, em vez de retrabalhar o aplicativo inteiro.

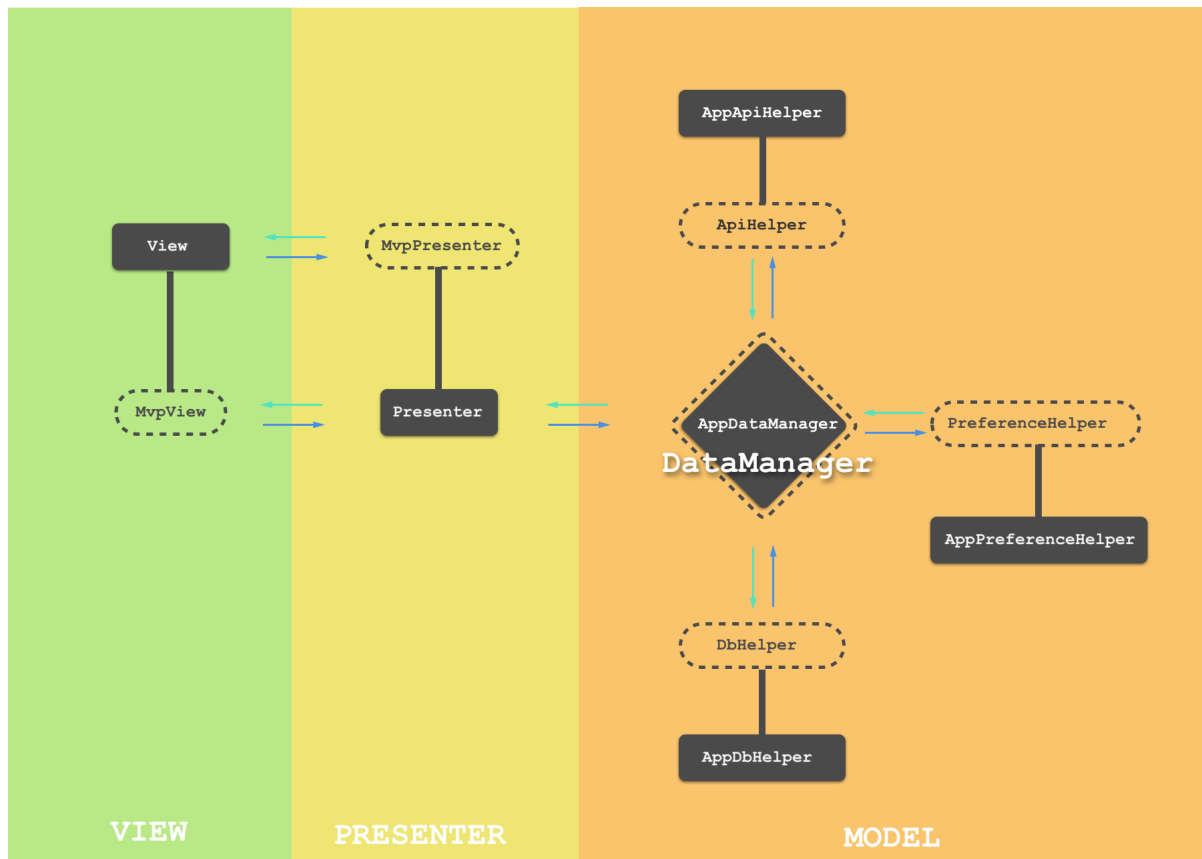
A Arquitetura em Camadas visa a criação de aplicativos modulares, de forma que a camada mais alta se comunica com a camada mais baixa e assim por diante, fazendo com que uma camada seja dependente apenas da camada imediatamente abaixo. ([BACALÁ, 2017](#))

Para implementar a arquitetura em camadas nesse projeto utilizamos o padrão arquitetural MVP (*Model-View-Presenter*). O escolhemos porque ele se adapta bem à estrutura do *framework* do Android. O MVP é um conjunto de diretrizes que, se seguidas, desacopla o código deixando-o reusável e testável. Este padrão divide a aplicação em componentes baseados em seu respectivo papel, prática conhecida como separação de conceitos ([ALI, 2017](#)). O MVP divide a aplicação em três componentes básicos:

- Model (Modelo): responsável pelo manuseio dos dados da aplicação.
- View (Visão): Responsável pela apresentação dos dados na tela da aplicação.
- Presenter (Apresentador): É a ponte que conecta a Model e a View, bem como age como um instrutor para a View.

Baseando-se no modelo sugerido por [Ali \(2017\)](#) apresentado na Figura 25, a arquitetura do *Smart Parking* foi elaborada.

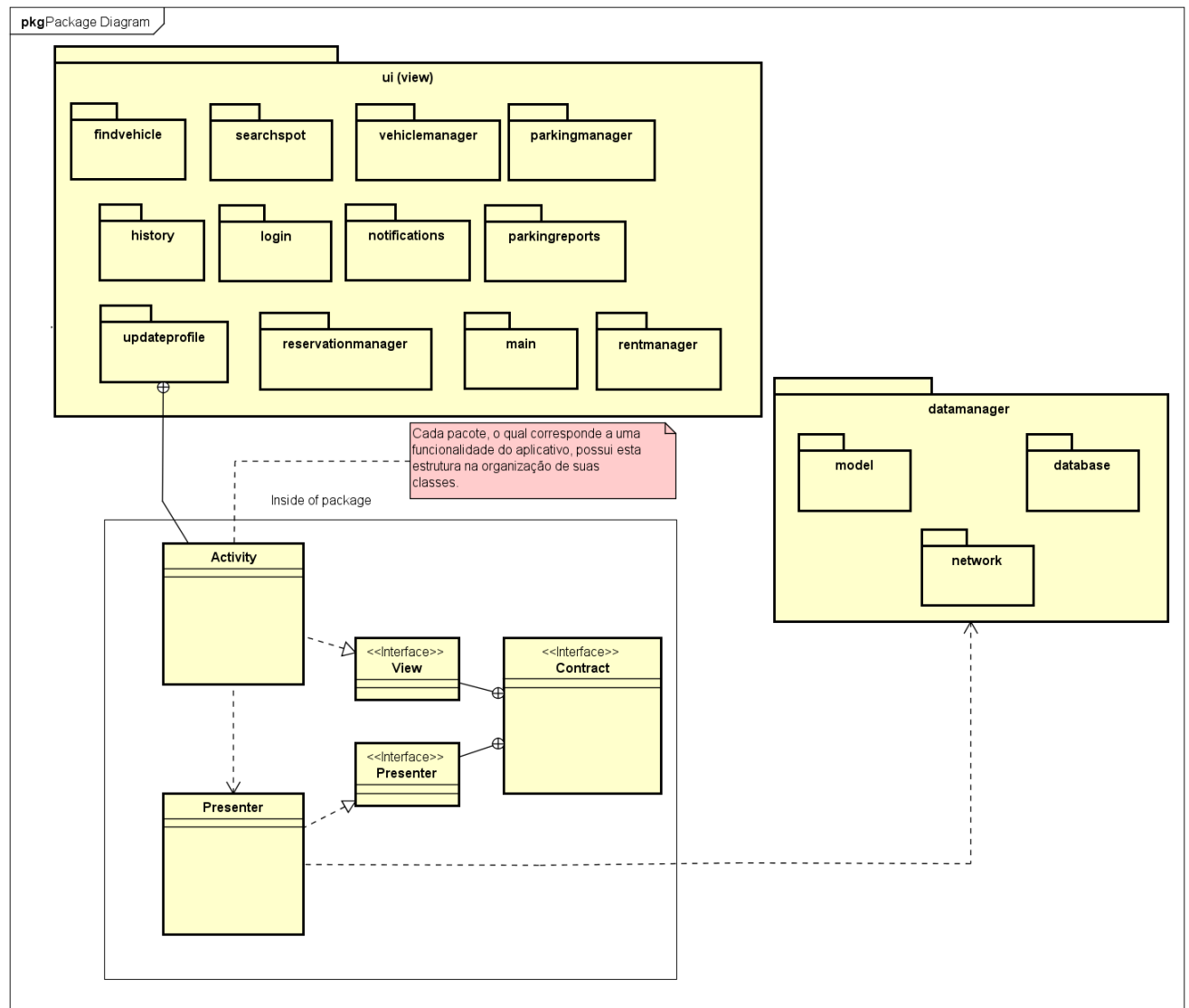
Figura 25 – Organização da Arquitetura seguindo as diretrizes MVP



Fonte: [Janishar Ali - Mindworks](#)

Para representação da arquitetura projetada foi utilizado um diagrama de Pacotes. Um diagrama de Pacotes é composto de pacotes (representações de um grupo de classes - ou outros elementos - que se relacionam com outros pacotes por meio de uma relação de dependência) e dos relacionamentos entre eles. A Figura 26 apresenta o diagrama de pacotes do *Smart Parking*. Neste, cada pacote que fica dentro do UI guarda a classe que implementa a *View*, assim como a classe que implementa o *Presenter* e uma classe que implementa o contrato do relacionamento entre as duas primeiras, ou seja, uma interface que descreve os métodos que devem ser implementados tanto pela *View* quanto pelo *Presenter*, por meio dos quais as duas classes irão se comunicar. Vale ressaltar que existe um pacote para cada funcionalidade do aplicativo, assim como um pacote que guarda todas as classes responsáveis pelo gerenciamento dos dados.

Figura 26 – Diagrama de Pacotes - Arquitetura do aplicativo proposto baseado no padrão MVP



Fonte: Próprios autores

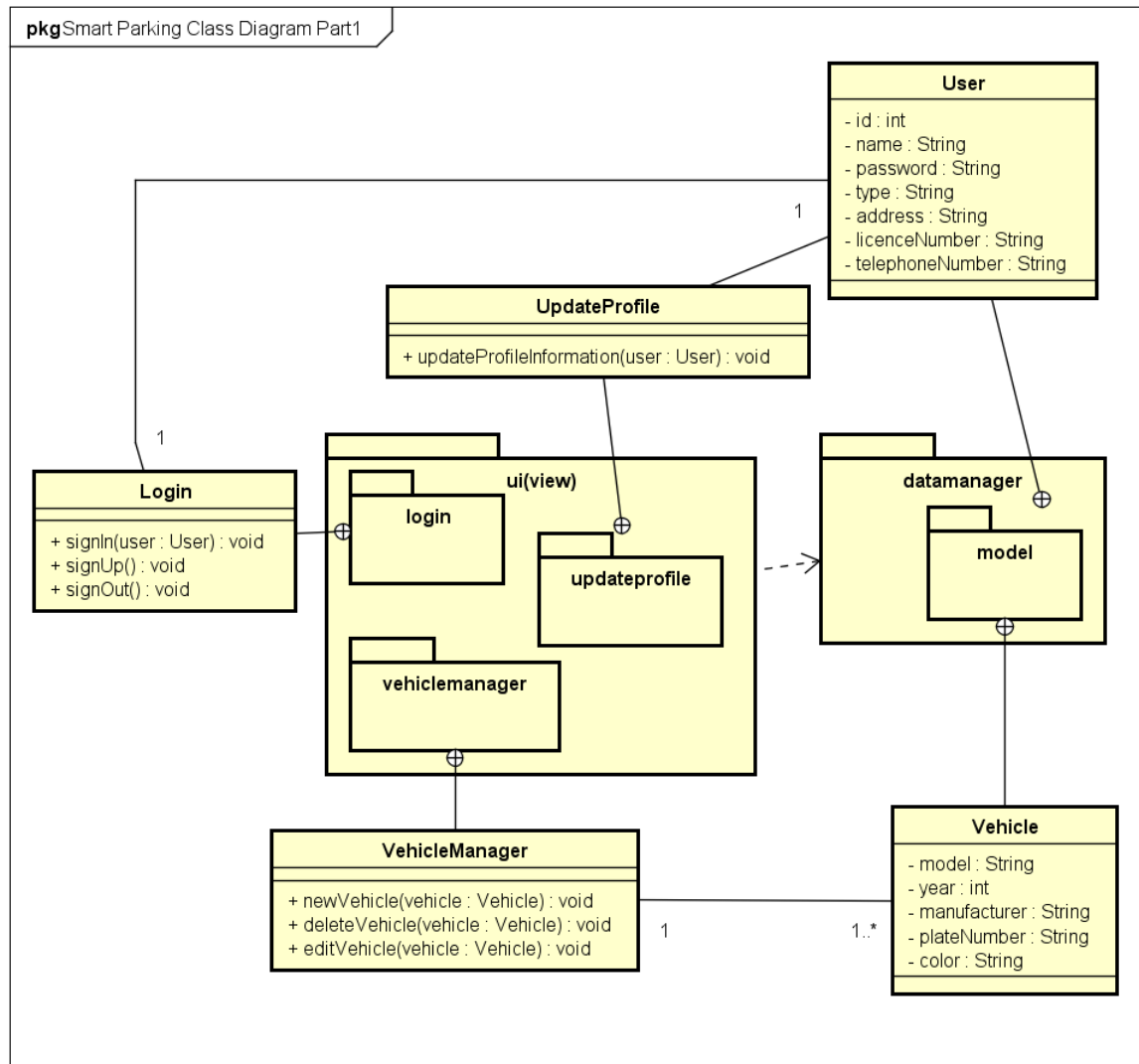
4.7 Diagrama de Classes

O Diagrama de Classes é uma representação da estrutura e das relações dos classificadores (usualmente classes, mas que também podem ser uma interface, um tipo de dado ou um componente) que servem de modelo para objetos. É um dos diagramas centrais da notação UML (BELL, 2016). Para facilitar a visualização e a compreensão do Diagrama de Classes e aproveitando o modelo apresentado pelo Diagrama de Pacotes (Figura 26) foi construído o Diagrama de Classes do Projeto, dividido em 4 partes, como apresentamos nas Figuras 27, 28, 29 e 30.

A Figura 27 ilustra as classes ligadas ao autenticação de usuários e a manutenção dos

dados básicos da aplicação, bem como seus relacionamentos.

Figura 27 – Diagrama de Classes - SmartParking (Parte 1)

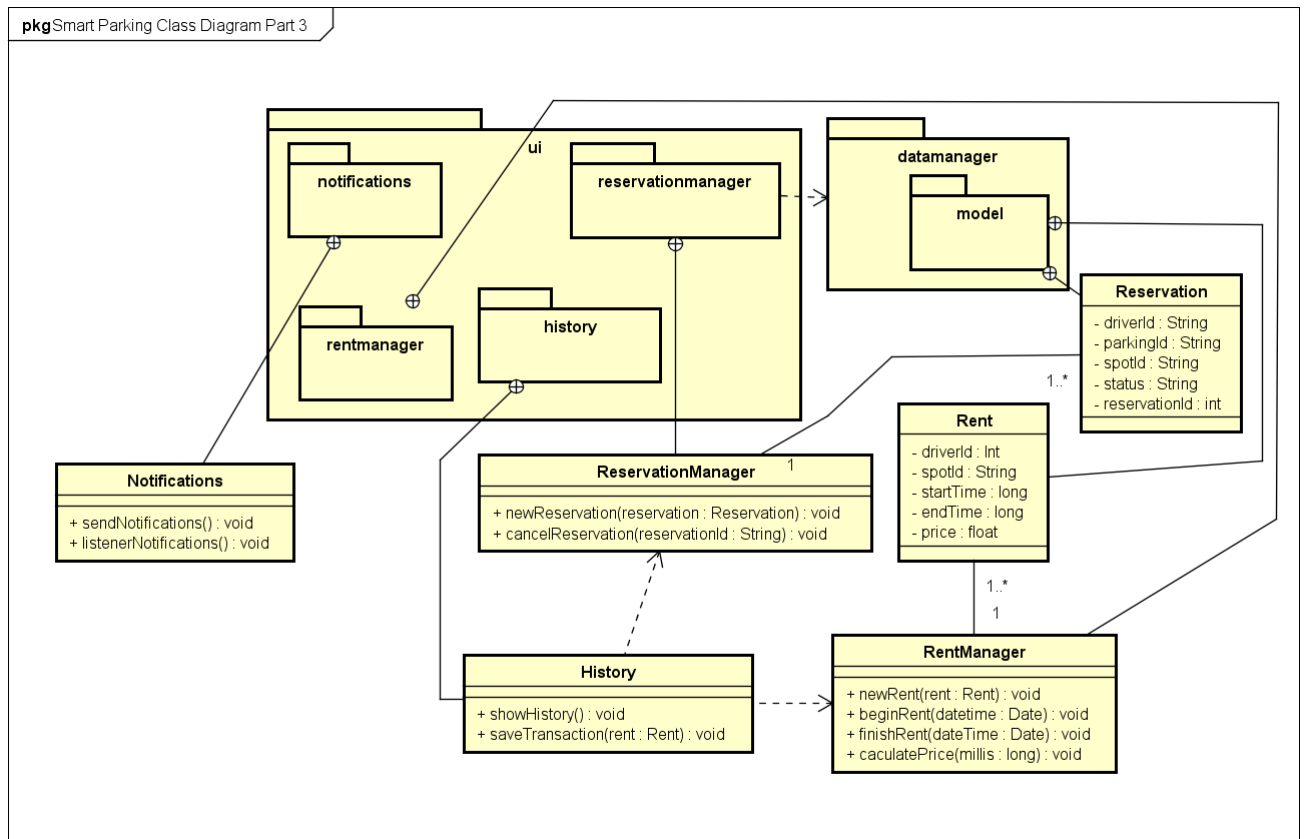


powered by Astah

Fonte: Próprios autores

A Figura 28 ilustra as classes ligadas à busca de vagas e seus relacionamentos, assim como as classes referentes ao gerenciamento do estacionamento.

Figura 29 – Diagrama de Classes - SmartParking (Parte 3)

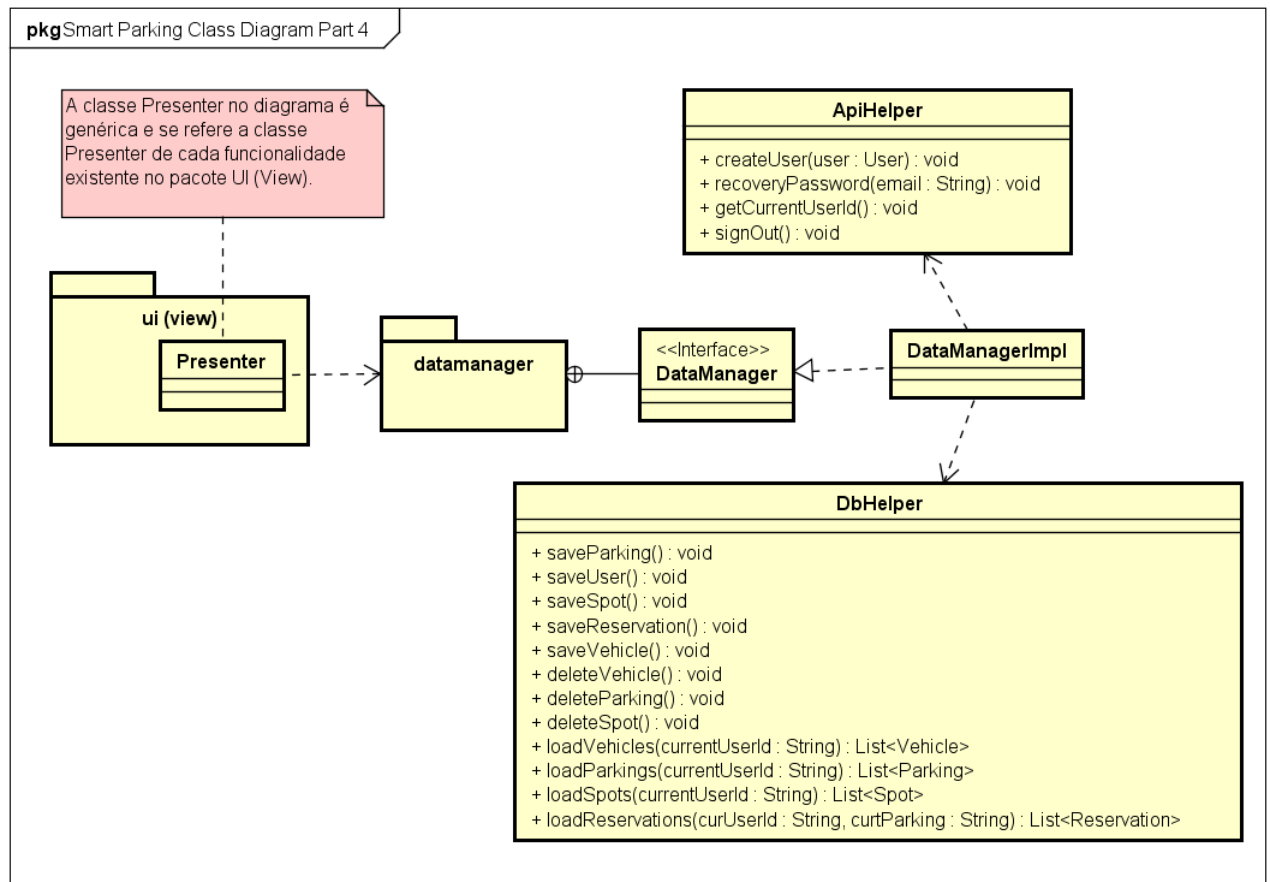


powered by Astah

Fonte: Próprios autores

A Figura 30 ilustra como as classes responsáveis por provê uma interface aos usuários se relacionam com àquelas responsáveis pelo gerenciamento dos dados na aplicação.

Figura 30 – Diagrama de Classes - SmartParking (Parte 4)



powered by Astah

Fonte: Próprios autores

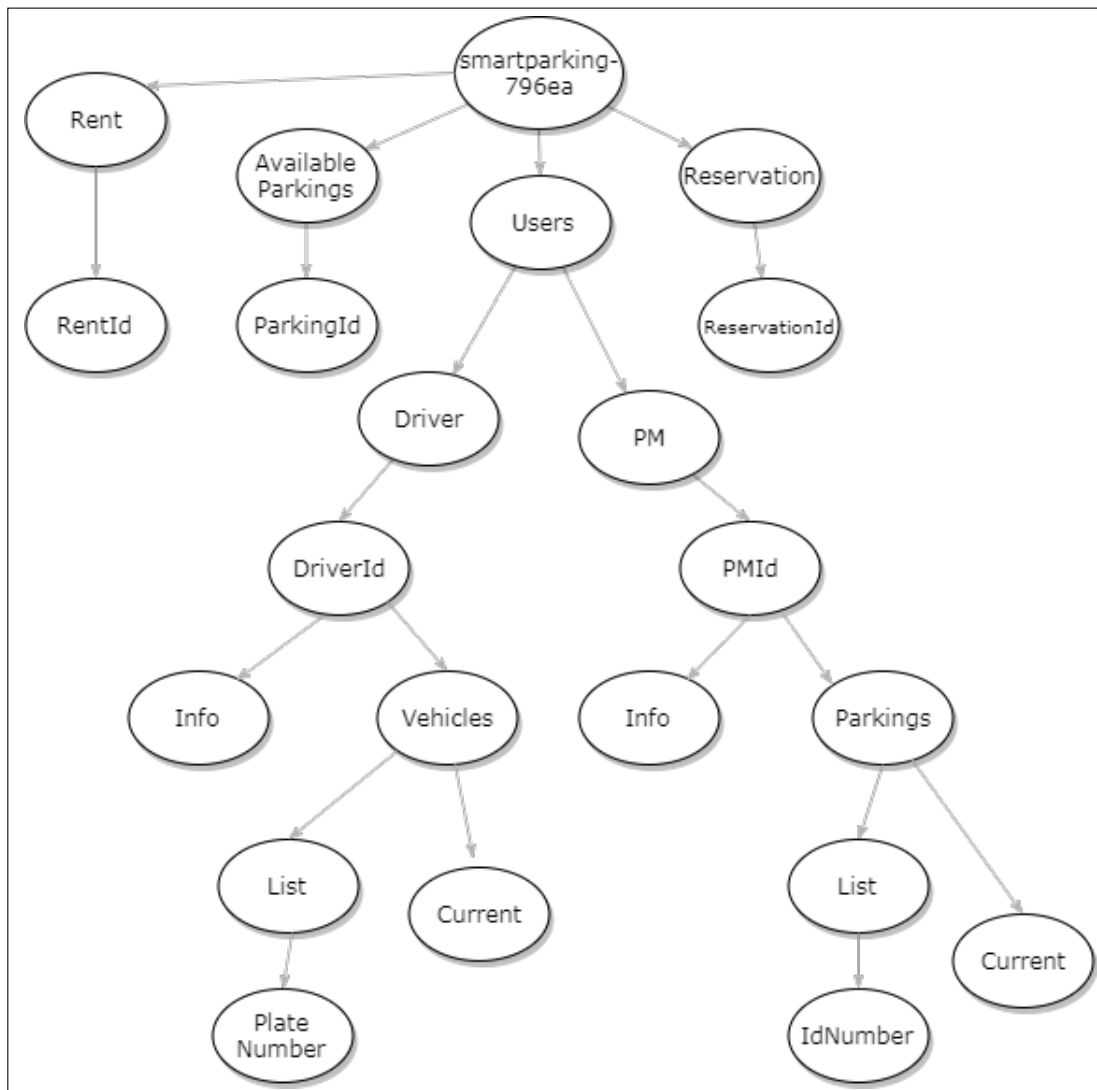
4.8 Esquema do Banco de Dados

Para a implementação do Banco de Dados foi escolhido o *Firebase Realtime Database*¹⁴, uma estrutura Json¹⁵ para armazenamento de dados dispostos em forma de árvore. Esta ferramenta foi escolhida devido a sua praticidade, fácil integração, disponibilidade e por, em um primeiro momento, ser uma ferramenta gratuita e de fácil acesso. A Figura 31 mostra a estrutura completa do banco de dados necessários para alimentar o *Smart Parking*.

¹⁴ Firebase Realtime Database: <https://firebase.google.com/docs/database/?hl=pt-br>

¹⁵ JSON (JavaScript Object Notation) é um formato leve de intercâmbio de dados (CROCKFORD, 2014).

Figura 31 – Modelo Json - Firebase Realtime Database



Fonte: Próprios autores

Cada folha (nós da extremidade inferior) representada na Figura 31 é referente a um registro (objeto) no Banco de Dados, estes são organizados a partir de um número de identificação que descreve o nó, sendo que os dados estão contidos neste último. Além do *Realtime Database*, o módulo de autenticação do *Firebase* também foi utilizado para o gerenciamento da autenticação do usuário.

4.9 Planejamento das Sprints

O planejamento das Sprints foi guiado pela estimativa do esforço gasto no desenvolvimento das histórias e pela definição das prioridades das mesmas. Ressalta-se que as histórias foram construídas a partir dos requisitos que foram levantados.

Para estimar o esforço despendido em cada história foi utilizada a estratégia de *Planning*

Poker. Planning Poker é uma técnica para estimação de esforço, baseada em consenso, cujo formato se assemelha a um jogo de cartas. Sua principal aplicação é para estimar o esforço gasto no desenvolvimento dos artefatos, tarefas ou histórias no contexto do desenvolvimento de softwares. E é rotineiramente utilizada em conjunto com o Scrum.

No *Planning Poker*, para cada história ou tema a ser estimado, um moderador (geralmente o *Product Owner* ou um analista) lê a descrição. Então, haverá alguma discussão, onde o *Product Owner* responderá a quaisquer perguntas que os estimadores tenham. Mas o objetivo do *Planning Poker* não é obter uma estimativa que resista a todo o futuro escrutínio. Em vez disso, deseja-se obter uma estimativa valiosa que possa ser obtida de forma rápida. Após a discussão, cada estimador seleciona privadamente uma carta com um valor que representa sua estimativa.

Uma vez que cada estimador fez uma seleção, as cartas são simultaneamente entregues e mostrados para que todos os participantes possam ver as estimativas uns dos outros. Os estimadores mais altos e mais baixos explicam suas perspectivas para que a equipe possa saber de onde elas vieram. O moderador pode tomar notas durante esta sessão de planejamento que poderão ser úteis quando a história for programada e testada.

Após a discussão, cada estimador realiza uma nova estimativa. Muitas vezes, as estimativas convergem na segunda rodada. Caso contrário, repete-se o processo até que a equipe concorde com uma única estimativa para a história. Raramente leva-se mais de três rodadas para alcançar o objetivo (COHN, 2017).

Ao esconder as cartas durante a estimação individual, o grupo pode evitar o viés cognitivo da ancoragem, onde o primeiro número falado em voz alta estabelece um precedente para as estimativas subsequentes.

A sequência numerada de valores utilizada nas cartas pode variar, mas, costuma-se utilizar a sequência de Fibonacci¹⁶(ou sequências baseadas nela). A principal razão para isso é a facilidade de dimensionar o tamanho das tarefas utilizando essa sequência, já que há uma distinção clara entre as quantidades representadas pelos números e essa distinção se acentua à medida em que se avança na sequência.

Para realizar o *Planning Poker* utilizamos a estrutura disponibilizada pela ferramenta online *Firepoker*¹⁷. Inicialmente foram adicionadas as histórias. Após isso, cada história começou a ser estimada em uma série de rodadas em que cada autor privadamente selecionava um valor para a tarefa entre aqueles apresentados nos cartões disponibilizados pelo site (a sequência de valores nos cartões utilizados pelo site era a dos múltiplos de 2). Quando não havia um consenso após a primeira rodada, os autores conversavam sobre a tarefa e os cartões eram selecionados novamente. Se ainda houvesse divergência era obtida uma média aritmética dos

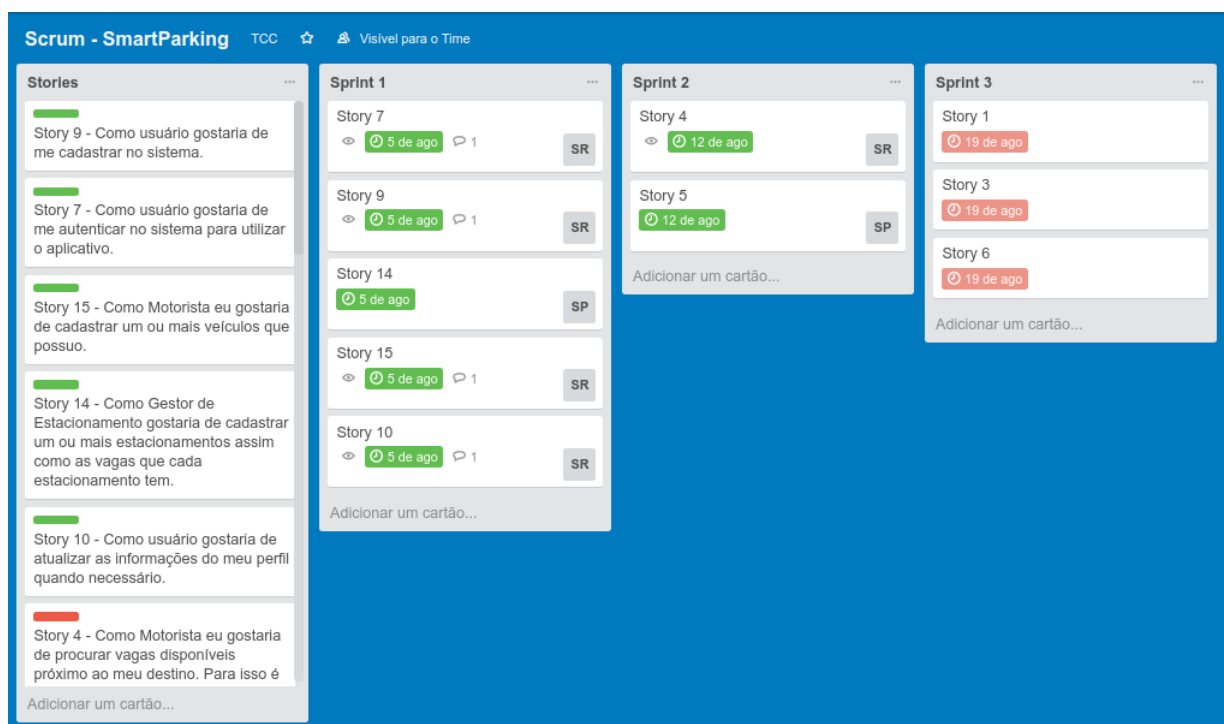
¹⁶ A sequência de Fibonacci é uma sequência de números inteiros, começando normalmente por 0 e 1, na qual, cada termo subsequente corresponde à soma dos dois anteriores. Recebeu esse nome em homenagem ao matemático Leonardo Fibonacci, que a descreveu em 1202 d.C.

¹⁷ O site do Firepoker pode ser acessado por meio do link: <http://firepoker.io/>

valores selecionados e essa média era atribuída a história.

Com as histórias estimadas, foi realizada a distribuição das mesmas nas *sprints*. Nas primeiras *sprints* foram adicionadas as histórias que continham os requisitos mais essenciais ao aplicativo, dando preferência para aqueles que possuíam menor nível de dificuldade. Nas *sprints* seguintes foram introduzidos os requisitos essenciais que continham grau de dificuldade maior. Por fim, nas últimas *sprints* foram alocadas as histórias que continhas os requisitos desejados, mas não essenciais. A Figura 32 ilustra a distribuição das histórias, com suas respectivas estimações (representadas por etiquetas do Trello em cores diferentes), ao longo das *sprints*.

Figura 32 – Tela do Trello - Planejamento das Sprints



Fonte: Próprios autores

As 15 histórias elaboradas foram divididas em 6 Sprints e destas, 2 Sprints foram cumpridas totalizando 7 histórias implementadas, sendo elas os requisitos RF01, RF02, RF05, RF08, RF10 e RF11. Mais informações sobre o aplicativo estão no manual do usuário, disponível no Apêndice G.

4.10 Implementação do Protótipo do Produto

Esta seção descreve o processo de implementação do protótipo do produto. Este protótipo é composto pelas funcionalidades descritas nas 7 histórias (descritas no Quadro 23) que compõem as duas primeiras *sprints*, as quais foi possível implementar no tempo disponível.

Quadro 23: Histórias Implementadas

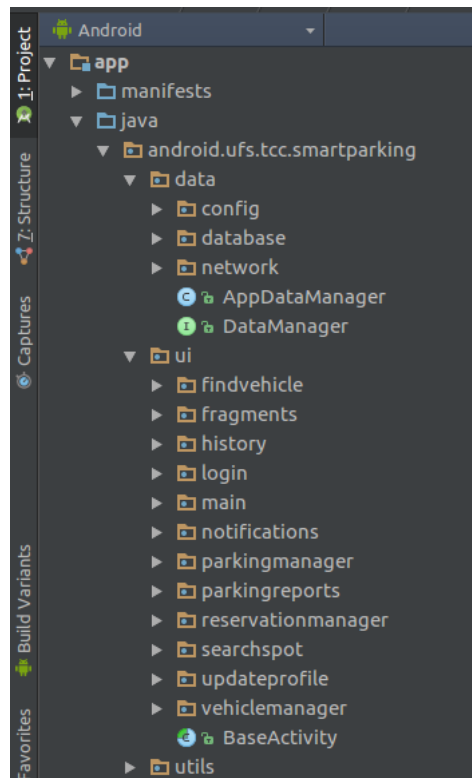
Nº da História	Descrição
Story 9	Como usuário gostaria de me cadastrar no sistema.
Story 7	Como usuário gostaria de me autenticar no sistema para utilizar o aplicativo.
Story 15	Como Motorista eu gostaria de cadastrar um ou mais veículos que possuo.
Story 14	Como Gestor de Estacionamento gostaria de cadastrar um ou mais estacionamentos assim como as vagas que cada estacionamento tem.
Story 10	Como usuário gostaria de atualizar as informações do meu perfil quando necessário.
Story 4	Como Motorista eu gostaria de procurar vagas disponíveis próximo ao meu destino. Para isso é necessária uma lista de vagas disponíveis nos estacionamentos cadastrados próximos a localidade informada.
Story 5	Como Motorista eu gostaria de reservar uma vaga. Para isso é preciso cadastrar a reserva e mudar o status da vaga.

Com o objetivo de elaborar um código fácil de manter e testável, além de obedecer os princípios da arquitetura mencionada na Seção 4.6, também foram adotadas algumas boas práticas, como a utilização dos padrões de projeto *Singleton*, *Observer* e *Adapter*, a convenção da nomenclatura dos atributos, métodos e classes em estruturas predefinidas e escrita do código em língua inglesa e a atualização das versões do projeto por meio de um sistema de controle de versão.

Uma vez que foram utilizados serviços de terceiros, foi necessário importar as bibliotecas referentes a estes serviços, entre elas as bibliotecas do Firebase para a integração com o *RealTime Database* e com o módulo de autenticação e as bibliotecas da API de serviços da Google para ter acesso ao mapa do Android, necessário nesta aplicação.

Como mencionado na Arquitetura, o projeto foi organizado de forma que os pacotes fossem separados por funcionalidade (Figura 33), sendo que cada pacote contém 3 classes principais, uma *Activity* (componentes do *framework* do Android responsáveis por criar uma tela com a qual os usuários interagem), representando a *View*, o *Presenter* e a interface de contrato entre as duas primeiras classes. Além destas classes principais algumas funcionalidades demandaram a implementação de *Fragments* (partes de uma *Activity*) e *Adapters* (adaptadores para exibição de listas customizadas). Também foi criado o pacote data, o qual armazena todas as classes responsáveis por gerenciar os dados de entrada e saída da aplicação e o pacote util, no qual estão as classes de suporte. Os códigos 1, 2 e 3 exemplificam a estrutura principal mencionada.

Figura 33 – Árvore de diretórios do projeto



Fonte: Próprios autores

O Código 1 exemplifica a estrutura utilizada na implementação do contrato entre as classes da *View* e o *Presenter*.

Código 1 – Contract - Interface VehicleManagerContract

```
1 public interface VehicleManagerContract {
2     interface View {
3         void onVehicleSearchCompleted (Vehicle vehicle);
4     }
5     interface Presenter {
6         void saveVehicle(Vehicle vehicle, String currentUserId);
7         List<Vehicle> loadVehicles(String currentUserId);
8         void deleteVehicle(Vehicle vehicle, String currentUserId);
9         String getCurrentUserId();
10        void searchCurrentVehicle(String currentUserId);
11        void setCurrentVehicle (Vehicle vehicle, String currentUserId);
12    }
13 }
```

O Código 2 exemplifica a estrutura utilizada na implementação da Activity referente a funcionalidade de gerenciamento de veículos, a qual representa a *View* desta funcionalidade.

Código 2 – View - VehicleManagerActivity

```
1 public class VehicleManagerActivity extends AppCompatActivity implements
2     ↪ VehicleManagerContract.View, View.OnClickListener,
3     ListView.OnItemClickListener, VehicleListFragment.OnVehicleClick{
4
5     private FloatingActionButton fabAdd;
6     private VehicleManagerContract.Presenter mVehicleManagerPresenter;
7     private Toolbar toolbar;
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_vehicle_manager);
13
14        mVehicleManagerPresenter = new VehicleManagerPresenter(this);
15    }
16
17    @Override
18    public void clickedOnVehicle(Vehicle vehicle) {
19        mVehicleManagerPresenter.setCurrentVehicle(vehicle,
20            ↪ mVehicleManagerPresenter.getCurrentUserId());
21        finish();
22    }
23
24    @Override
25    public void onVehicleSearchCompleted(Vehicle vehicle) {
26    }
```

O Código 3 exemplifica a estrutura utilizada na implementação do *Presenter*, camada intermediária que separa a *View* do módulo responsável pelo gerenciamento dos dados.

Código 3 – Presenter - VehicleManagerPresenter

```
1  public class VehicleManagerPresenter implements VehicleManagerContract.Presenter{
2
3      private DataManager mDataManager = new AppDataManager();
4      private VehicleManagerContract.View mManagerView;
5
6      public VehicleManagerPresenter(VehicleManagerContract.View managerView) {
7          mManagerView = managerView;
8      }
9
10     @Override
11     public void saveVehicle(Vehicle vehicle, String currentUserId) {
12         mDataManager.saveVehicle(vehicle, currentUserId);
13     }
14
15     @Override
16     public List<Vehicle> loadVehicles(String currentUserId) {
17         return mDataManager.loadVehicles(currentUserId);
18     }
19
20     @Override
21     public void deleteVehicle(Vehicle vehicle, String currentUserId) {
22         mDataManager.deleteVehicle(vehicle, currentUserId);
23     }
24
25     @Override
26     public String getCurrentUserId() {
27         return mDataManager.getCurrentUser();
28     }
29
30     @Override
31     public void searchCurrentVehicle(String currentUserId) {...}
32
33     @Override
34     public void setCurrentVehicle(Vehicle vehicle, String currentUserId) {
35         mDataManager.setCurrentVehicle( vehicle, currentUserId );
36     }
37
38
39 }
```

Vale ressaltar que otimizações podem ser realizadas no código posteriormente, como a utilização de *framework* para a injeção de dependência, visando alcançar um código cada vez mais coeso e menos acoplado.

4.11 Validação do Aplicativo

A Validação do Software é uma fase crucial no processo de desenvolvimento, uma vez que tem como objetivo certificar que este atende as necessidades do usuário e se de fato entrega o que promete. Visando validar as funcionalidades implementadas, o protótipo do aplicativo foi submetido a testes realizados por possíveis usuários, os quais responderam a um questionário após o uso da ferramenta e testes de caixa-preta realizados por alunos de Ciência da Computação da Universidade Federal de Sergipe.

4.11.1 Questionários de Validação

Para a validação do aplicativo desenvolvido, foram elaborados dois questionários, um para gestores de estacionamento (está disponível no Apêndice D) e outro para motoristas (está disponível no Apêndice E) com perguntas relacionadas às funcionalidades implementadas. Diferentes dos demais questionários distribuídos por intermédio de redes sociais, este foi respondido diretamente com o público alvo (motoristas e funcionários de estacionamentos da cidade de Aracaju), neste caso cada um dos respondentes utilizou o aplicativo e a partir da experiência com o mesmo respondeu as questões mediante uma entrevista direta.

Foram visitados 4 estacionamentos no centro de Aracaju/SE e entrevistados 4 funcionários um de cada um destes estabelecimentos. Conversando diretamente com os entrevistados foi possível perceber que alguns estacionamentos da cidade, a maioria entre os visitados, não possui muitos meios para o gerenciamento dos mesmos, enquanto alguns utilizam softwares especializados em *desktops*, outros usam máquinas registradoras e até mesmo tabelas em papel. Uma das grandes dificuldades apontadas por um dos entrevistados é a pouca visibilidade destes estacionamentos que, segundo ele: "quase ninguém sabe da existência". Um outro dos entrevistados informou que outra grande desvantagem nos estacionamentos da região é que quase nenhum deles aceita cartão de crédito. Durante a entrevista, foi observado nos quatro estacionamentos que todo o processo de registro de entrada e saída dos veículos é feita manualmente.

Embora, dois dos entrevistados tenham sugerido que a reserva não é uma função interessante devido a rotatividade destes estacionamentos, o *feedback* dos entrevistados em relação ao aplicativo foi positivo, todos eles afirmaram que utilizariam o *Smart Parking* e foram dadas sugestões interessantes em relação ao aplicativo.

Os motoristas, por sua vez, foram entrevistados da mesma forma que os funcionários do estacionamento, de forma direta. A maioria dos escolhidos para responder o questionário possui experiência como motorista profissional e avaliou o aplicativo de acordo com as necessidades que os mesmos sentem no dia a dia. Sugestões também foram dadas em relação a cada funcionalidade do aplicativo.

Uma das dificuldades encontradas na aplicação deste questionário foi a restrição dos entrevistados devido ao fato de ser um tipo de usuário muito específico. No total foram conseguidas

10 respostas se considerados ambos os questionários juntos, portanto, sabemos da necessidade de mais entrevistas para uma validação mais acurada.

4.11.2 Testes Realizados

[Sommerville \(2007\)](#) sugere que para realização de testes posteriores aos de unidade (executados durante a implementação) uma equipe independente de testadores deve trabalhar com um plano de testes previamente elaborado. Para [Pressman \(2011\)](#) o papel de um grupo independente de testes é remover problemas associados ao fato de deixar o criador testar uma coisa que ele mesmo criou evitando conflito de interesses ou qualquer viés ou vício na execução destes testes. Seguindo esta recomendação foi escolhida uma equipe para a avaliação das funcionalidades do protótipo, prática também conhecida como teste de caixa-preta. O teste de caixa-preta analisa se todos os requisitos definidos foram aplicados no software e estão funcionando ([BLANCO, 2012](#)).

Os testes aplicados no *Smart Parking* foram realizados por 3 alunos do curso de Ciência da Computação da Universidade Federal de Sergipe e para a realização deste foi utilizado o plano de teste disponível no Apêndice F deste documento. O plano é composto pelos casos e procedimentos de testes, além das observações pontuadas pelos testadores na execução dos mesmos.

5

Considerações Finais e Trabalhos Futuros

Este trabalho apresenta uma aplicação móvel que procura otimizar os processos de oferta e procura de vagas de estacionamento, com o intuito de promover mais fluidez no tráfego de veículos e colaborar com o desenvolvimento de cidades inteligentes.

Para o desenvolvimento dessa aplicação: foi realizada uma revisão sistemática (com o intuito de descobrir qual o estado da arte na área de gestão inteligente de locais de estacionamento), uma revisão de produtos no mercado (com o objetivo de descobrir quais as características das aplicações que lidam com a gestão de vagas de estacionamento) e questionários de validação com possíveis usuários(para levantar quais as reais necessidades dos possíveis utilizadores da aplicação).

A Revisão Sistemática foi realizada para a extração de informações relevantes de trabalhos publicados em bases acadêmicas que tratam do gerenciamento inteligente de vagas de estacionamento, foram extraídas dela as características mais relevantes destes trabalhos.

A Pesquisa de Produtos no Mercado foi realizada com o objetivo de conhecer o que já existe nas lojas de aplicativos e quais são as características das aplicações disponibilizadas.

Posteriormente aplicou-se um questionário a fim de capturar a opinião de possíveis utilizadores da ferramenta e um protótipo funcional foi desenvolvido com o objetivo de validar as características da aplicação.

A partir do material extraído, foi realizado o planejamento das *sprints* e o desenvolvimento da aplicação em si. Após isso, a aplicação construída foi testada e, posteriormente, validada tanto com motoristas quanto com gestores de estacionamento.

Dos requisitos funcionais definidos foi possível concretizar nessa etapa: manter usuários (RF01), manter estacionamentos (RF02), manter vagas (RF05), manter reserva de vagas (RF08),

manter veículo (RF010), buscar vagas (RF011). Esses requisitos foram implementados prioritariamente devido à sua relevância para a proposta da aplicação e para o público-alvo da mesma. O tempo reduzido para o desenvolvimento da aplicação e a necessidade dos desenvolvedores de se familiarizarem com o desenvolvimento de uma aplicação móvel foram os principais fatores que impediram que mais requisitos fossem entregues nessa versão. Contudo os requisitos que não foram implementados e outros que surgiram ao longo do desenvolvimento, especialmente por meio das validações com os possíveis usuários, foram incorporados à lista de trabalhos futuros.

Como trabalhos futuros, pretende-se: incluir a possibilidade de efetuar pagamentos por meio da aplicação; aumentar a acessibilidade; listar para os motoristas quais as melhores opções para estacionar em seu local de destino, de acordo com os critérios que ele elencar; lançar uma versão nativa da aplicação para o iOS (sistema operacional para aplicações móvel da Apple) e registrar a propriedade intelectual gerada pelo desenvolvimento da aplicação.

Gerenciar os pagamentos já é um dos requisitos desejados, mas que ainda não foi implementado. Para implementá-lo, pretende-se realizar a integração com soluções de pagamento, como o PagSeguro¹ e o PayPal². O intuito disso é permitir que os motoristas possam não apenas efetuar as reservas pela aplicação, mas também pagar por elas, por meio da mesma. Além disso, para os gestores é uma forma de ter mais controle sobre os valores que movimenta e também a possibilidade de ofertar outros meios de pagamento, como cartão de crédito, para os seus clientes.

Uma outra característica planejada para a aplicação é exibir para o motorista, em uma lista ordenada de acordo com os seus critérios, quais são as melhores opções para ele estacionar em seu local de destino. Ao invés de apenas listar os estacionamentos com vagas disponíveis na região, a aplicação indicará quais as melhores escolhas, levando em consideração, preço, distância e até informações sobre o tráfego na cidade. Além disso, almeja-se acrescentar as informações dos meios de transporte público disponíveis na cidade e levá-los em conta para determinar quais as melhores opções para o motorista. Isso significa que, ao invés de apenas listar estacionamentos próximos ao destino final do motorista, a aplicação irá sugerir também estacionamentos que fiquem em pontos intermediários, mas que sejam próximos de estações de metrô ou de ônibus, cujas linhas levam a região do destino final do motorista, desde que essa mudança de meio de transporte seja vantajosa para o motorista, segundo os critérios que ele selecionou.

Para aumentar a acessibilidade da aplicação, será adicionada a possibilidade de acessar as funcionalidades por comandos de voz. Será incluído também a opção de alterar o contraste entre as cores. Isso, com o intuito de facilitar a utilização da aplicação por pessoas com deficiências visuais.

Para ampliar o alcance da aplicação, esta terá uma versão desenvolvida para o iOS,

¹ Pagseguro: <https://pagseguro.uol.com.br/>

² PayPal: <https://www.paypal.com/br/home>

já que ele é o segundo sistema operacional móvel mais utilizado no mundo, juntos Android e o iOS correspondem a 97.87 % (SHARE, 2017) do mercado. Assim sendo, essa versão da aplicação para o S.O. móvel da Apple garantirá um alcance praticamente universal para a solução construída.

Por fim será registrada a propriedade intelectual gerada no desenvolvimento da aplicação junto ao INPI (Instituto Nacional da Propriedade Industrial)³.

³ INPI: <http://www.inpi.gov.br/>

Referências

- ABIDI, S. et al. A new heuristic for solving the parking assignment problem. *19th International Conference on Knowledge Based and Intelligent Information and Engineering Systems*, p. 312–321, 2015. Citado 2 vezes nas páginas 37 e 42.
- ALI, J. *Essential Guide For Designing Your Android App Architecture: MVP: Part 1*. 2017. Disponível em: <<https://blog.mindorks.com/essential-guide-for-designing-your-android-app-architecture-mvp-part-1-74efaf1cda40>>. Citado na página 71.
- ARELLANO-VERDEJO, J.; ALBA, E. Optimal allocation of public parking slots using evolutionary algorithms. In: *2016 International Conference on Intelligent Networking and Collaborative Systems (INCoS)*. [S.l.: s.n.], 2016. p. 222–228. Citado 2 vezes nas páginas 37 e 40.
- ATIF, Y.; DING, J.; JEUSFELD, M. Internet of things approach to cloud-based smart car parking. In: . [S.l.: s.n.], 2016. v. 58, p. 193–198. Cited By 0. Citado 2 vezes nas páginas 37 e 43.
- AUDENHOVE, F.-J. V. et al. *The Future of Urban Mobility 2.0*. [S.l.]: Arthur D. Little, 2014. Citado na página 22.
- BACALÁ, S. *Arquitetura em Camadas - FACOM - UFU*. 2017. Disponível em: <<http://www.facom.ufu.br/~bacala/PI/WebCamadas.pdf>>. Citado na página 71.
- BELL, D. *IBM - developerWorks - Fundamentos básicos de UML*. 2016. Disponível em: <<https://www.ibm.com/developerworks/br/rational/library/content/RationalEdge/sep04/bell/index.html>>. Citado na página 73.
- BERNARDO, K. *Como funciona o Scrum? - Cultura Ágil*. 2015. Disponível em: <<https://www.culturaagil.com.br/como-funciona-o-scrum/>>. Citado na página 26.
- BLANCO, M. Z. Documentação de teste baseado na norma iee 829 – estudo de caso: “sistema de apoio a tomada de decisão”. *T.I.S. São Carlos*, p. 91–97, jul 2012. ISSN 2316-2872. Citado na página 86.
- BUTOWSKY, A. et al. City of white plains parking app: Case study of a smart city web application. In: *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*. [S.l.: s.n.], 2015. p. 278–282. Citado 2 vezes nas páginas 37 e 40.
- CARAGLIU, A.; BO, C. D.; NIJKAMP, P. Smart cities in europe. 2009. Citado na página 22.
- CASTELLI, I. *Trello: como esta ferramenta pode ajudar você a organizar a sua vida | Tecmundo*. 2015. Disponível em: <<https://www.tecmundo.com.br/organizacao/75128-trello-ferramenta-ajudar-voce-organizar-vida.htm>>. Citado na página 63.
- CHEN, X.; LIU, N. Smart parking by mobile crowdsensing. *International Journal of Smart Home*, 2016. Citado 2 vezes nas páginas 37 e 44.

- COHN, M. *Planning Poker Cards: Effective Agile Planning and Estimation*. 2017. Disponível em: <<https://www.mountaingoatsoftware.com/tools/planning-poker>>. Citado na página 79.
- CROCKFORD, D. *The JavaScript Object Notation (JSON) Data Interchange Format*. 2014. Disponível em: <<https://tools.ietf.org/pdf/rfc7159.pdf>>. Citado na página 77.
- CUNHA, M. A. et al. *SMART CITIES: TRANSFORMAÇÃO DIGITAL DE CIDADES*. FGV-EAESP, 2016. Disponível em: <http://cta.ornl.gov/data/tedb35/Edition35_Full_Doc.pdf>. Citado 2 vezes nas páginas 18 e 21.
- DAVIS, S. C.; WILLIAMS, S. E.; BOUNDY, R. G. *Transportation Energy Data Book*. U.S. DEPARTMENT OF ENERGY, 2016. Disponível em: <http://cta.ornl.gov/data/tedb35/Edition35_Full_Doc.pdf>. Citado na página 18.
- DI NOCERA; DI NAPOLI, C. D.; ROSSI, S. A social-aware smart parking application. 2015. Citado 2 vezes nas páginas 37 e 42.
- DOBIE, A. et al. *Android Central*. Disponível em: <<http://www.androidcentral.com/android-pre-history>>. Citado 2 vezes nas páginas 27 e 28.
- DUCROHET, X.; NORBYE, T.; CHOU, K. *Android Studio: A IDE built for Android*. 2013. Disponível em: <<https://android-developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html>>. Citado na página 61.
- DUNN, J. *There's no hope of anyone catching up to Android and iOS* | *Business inside*. 2016. Disponível em: <<http://www.businessinsider.com/smartphone-market-share-android-ios-windows-blackberry-2016-8>>. Citado na página 28.
- ENGEL, G. T.; SILVEIRA, D. T. Métodos de pesquisa. *coordenado pela Universidade Aberta do Brasil – UAB/UFRGS e pelo Curso de Graduação Tecnológica – Planejamento e Gestão para o Desenvolvimento Rural da SEAD/UFRGS. – Porto Alegre: Editora da UFRGS*, p. 120, 2009. Citado na página 20.
- ESPAÑA, S. M. *Telpark - Apps para Android no Google Play*. 2017. Disponível em: <<https://play.google.com/store/apps/details?id=com.delaware.empark&hl=pt>>. Citado na página 52.
- FARKAS, K.; LENDÁK, I. Simulation environment for investigating crowd-sensing based urban parking. In: *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. [S.l.: s.n.], 2015. p. 320–327. Citado 2 vezes nas páginas 37 e 43.
- FEHER, G.; LAJTHA, B. A.; LOVASZ, A. A probabilistic approach to parking: Benefits of routing instead of spotting. In: *2015 International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*. [S.l.: s.n.], 2015. p. 1–6. Citado 2 vezes nas páginas 36 e 39.
- FONSECA, J. J. S. Metodologia da pesquisa científica. 2002. Citado na página 20.
- GARCIA, M. et al. Secure smart parking at james madison university via the cloud environment (space). In: . [S.l.: s.n.], 2016. p. 271–276. Citado 2 vezes nas páginas 37 e 41.
- GIL, A. C. Como elaborar projetos de pesquisa. *São Paulo:Atlas*, v. 4, 2007. Citado na página 20.

- GOOGLE. *Recursos do Android Studio*. 2017. Disponível em: <<https://developer.android.com/studio/features.html?hl=pt-br>>. Citado na página 62.
- GRODI, R.; RAWAT, D. B.; RIOS-GUTIERREZ, F. Smart parking: Parking occupancy monitoring and visualization system for smart cities. 2016. Citado 2 vezes nas páginas 36 e 38.
- IEEE. Systems and software engineering – vocabulary. *ISO/IEC/IEEE 24765:2010(E)*, p. 1–418, Dec 2010. Citado na página 23.
- IEEE. *About - IEEE Smart Cities*. 2014. Disponível em: <<http://smartcities.ieee.org/about.html>>. Citado na página 22.
- INRIX, I. *ParkMe - Apps para Android no Google Play*. 2017. Disponível em: <https://play.google.com/store/apps/details?id=com.parkme.consumer&hl=pt_BR>. Citado na página 54.
- INTERNACIONAL, P. *ParkApp Admin Estac. e Valets - Apps para Android no Google Play*. 2016. Disponível em: <<https://play.google.com/store/apps/details?id=com.ionicframework.parkapp>>. Citado na página 48.
- Jl, Z. et al. A cloud-based car parking middleware for iot-based smart cities: Design and implementation. *Sensors*, 2014. Citado 2 vezes nas páginas 37 e 42.
- Jl, Z. et al. A cloud-based intelligent car parking services for smart cities. In: *2014 XXXIth URSI General Assembly and Scientific Symposium (URSI GASS)*. [S.l.: s.n.], 2014. p. 1–4. Citado 2 vezes nas páginas 36 e 39.
- JUNIOR, L. A. da S. *App stores: number of apps in leading app stores 2017* | Statista. 2003. Disponível em: <<https://www.vivaolinux.com.br/artigo/O-movimento-do-codigo-aberto>>. Citado na página 27.
- KHANNA, A.; ANAND, R. Iot based smart parking system. In: *2016 International Conference on Internet of Things and Applications (IOTA)*. [S.l.: s.n.], 2016. p. 266–270. Citado 2 vezes nas páginas 36 e 38.
- KITCHENHAM, B. Procedures for performing systematic reviews. In: *Technical report, EBSE Technical Report. EBSE*. [S.l.]: sn, 2004. Citado 2 vezes nas páginas 29 e 30.
- KRIEG, J. G. et al. Unlocking the smartphone's senses for smart city parking. In: *2016 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2016. p. 1–7. Citado 2 vezes nas páginas 37 e 40.
- LAMBRINOS, L.; DOSIS, A. Applying mobile and internet of things technologies in managing parking spaces for people with disabilities. *UbiComp 2013*, 2013. Citado 2 vezes nas páginas 37 e 43.
- LAMBRINOS, L.; DOSIS, A. Disassist: An internet of things and mobile communications platform for disabled parking space management. *Globecom 2013 - Symposium on Selected Areas in Communications*, 2013. Citado 2 vezes nas páginas 36 e 38.
- LIKERT, R. A technique for the measurement of attitudes. *Archives of psychology*, 1932. Citado na página 56.

MAINETTI, L. et al. A smart parking system based on iot protocols and emerging enabling technologies. In: *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. [S.l.: s.n.], 2015. p. 764–769. Citado 2 vezes nas páginas 37 e 40.

MANJOO, F. *A Murky Road Ahead for Android, Despite Market Dominance* | *New York Times*. 2015. Disponível em: <https://www.nytimes.com/2015/05/28/technology/personaltech/a-murky-road-ahead-for-android-despite-market-dominance.html?_r=0>. Citado 2 vezes nas páginas 27 e 28.

MARTINO, S. D.; ROSSI, S. An architecture for a mobility recommender system in smart cities. In: . [S.l.: s.n.], 2016. v. 58, p. 425–430. Cited By 0. Citado 2 vezes nas páginas 37 e 41.

MEJRI, N. et al. Reservation-based multi-objective smart parking approach for smart cities. In: *2016 IEEE International Smart Cities Conference (ISC2)*. [S.l.: s.n.], 2016. p. 1–6. Citado 2 vezes nas páginas 36 e 38.

NAKAMURA, M.; BOUSQUET, L. D. Constructing execution and life-cycle models for smart city services with self-aware iot. In: *2015 IEEE International Conference on Autonomic Computing*. [S.l.: s.n.], 2015. p. 289–294. Citado 2 vezes nas páginas 37 e 43.

ONDEPARAR. *OndeParar - Apps para Android no Google Play*. 2016. Disponível em: <<https://play.google.com/store/apps/details?id=br.com.ondeparar>>. Citado na página 51.

ORRIE, O.; SILVA, B.; HANCKE, G. A wireless smart parking system. *IECON2015-Yokohama*, p. 9–12, 2015. Citado 2 vezes nas páginas 36 e 38.

PARALI. *ParAli Estacionamento - Apps para Android no Google Play*. 2014. Disponível em: <<https://play.google.com/store/apps/details?id=br.com.estacionamento.parali&hl=pt>>. Citado na página 47.

POLYCARPOU, E.; LAMBRINOS, L.; PROTOPAPADAKIS, E. Smart parking solutions for urban areas. In: *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. [S.l.: s.n.], 2013. p. 1–6. Citado 2 vezes nas páginas 37 e 40.

PRADESH, D. of I. T. H. *Himachal Parking - Apps para Android no Google Play*. 2016. Disponível em: <https://play.google.com/store/apps/details?id=findparking.hp.dit.himachal.com.shimlaparking&hl=pt_BR>. Citado na página 50.

PRADO, E. P. *Conheça a história do Android, o sistema operacional mobile da Google* | *Superinteressante*. 2017. Disponível em: <<http://super.abril.com.br/galeria/conheca-a-historia-do-android-o-sistema-operacional-mobile-da-google/>>. Citado na página 27.

PRESS; BRAZIL information team of the Delegation to. *União Europeia apresenta exemplos de sucesso de cidades inteligentes durante Encontro dos Municípios com o Desenvolvimento Sustentável*. 2017. Disponível em: <<https://eeas.europa.eu/delegations/brazil/>>. Citado na página 18.

PRESSMAN, R. *Engenharia de Software*. AMGH, 2011. ISBN 9788563308337. Disponível em: <<https://books.google.com.br/books?id=eRIOuQAACAAJ>>. Citado 5 vezes nas páginas 23, 24, 25, 63 e 86.

REPOSITORY, G. H. O. data. *Registered vehicles - data by country*. 2015. Disponível em: <<http://apps.who.int/gho/data/node.main.A995>>. Citado na página 18.

ROSSETTI, R. J. F. *[IEEE Xplore] Readings on Smart Cities*. 2016. Disponível em: <http://smartcities.ieee.org/articles-publications/ieee-xplore-readings-on-smart-cities/february-2016.html>. Citado na página 22.

ROY, A. et al. Smart traffic parking management using iot. In: *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. [S.l.: s.n.], 2016. p. 1–3. Citado 2 vezes nas páginas 37 e 41.

SCHWABER, K.; SUTHERLAND, J. Guia do scrum. *Línea*). Consultado em, v. 12, 2013. Citado na página 25.

SHANG, D.; DOULET, J.-F.; KEANE, M. *Urban Informatics in China: Exploring the Emergence of the Chinese City 2.0*. [S.l.]: Marcus Foth, 2009. Citado na página 22.

SHARE, N. M. *Mobile/Tablet Operating System Market Share | Net Market Share*. 2017. Disponível em: <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=8&qpcustomid=1>. Citado 2 vezes nas páginas 28 e 89.

SOMMERVILLE, I. *Software Engineering*. Addison-Wesley, 2007. (International computer science series). ISBN 9780321313799. Disponível em: <https://books.google.com.br/books?id=B7idKfL0H64C>. Citado 4 vezes nas páginas 23, 24, 63 e 86.

SROTYR, M. et al. Prediction system of occupancy of parking spaces. In: *2015 Smart Cities Symposium Prague (SCSP)*. [S.l.: s.n.], 2015. p. 1–4. Citado 2 vezes nas páginas 36 e 39.

STATISTA. *App stores: number of apps in leading app stores 2017 | Statista*. 2017. Disponível em: <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>. Citado na página 26.

STRASSER, M.; MAUSER, D.; ALBAYRAK, S. Mitigating traffic problems by integrating smart parking solutions into an interconnected ecosystem. In: *2016 IEEE Symposium on Computers and Communication (ISCC)*. [S.l.: s.n.], 2016. p. 32–37. Citado 2 vezes nas páginas 37 e 44.

TAHERKHANI, M. et al. Blueparking: An iot based parking reservation service for smart cities. In: . [S.l.: s.n.], 2016. v. 24-25-May-2016, p. 86–88. Citado 2 vezes nas páginas 37 e 41.

TALENTAPPS. *Parking: Localizar meu carro - Apps para Android no Google Play*. 2017. Disponível em: <https://play.google.com/store/apps/details?id=il.talent.parking&hl=pt-br>. Citado na página 46.

TAVEIRA, F. *Parkingaki - Apps para Android no Google Play*. 2016. Disponível em: https://play.google.com/store/apps/details?id=pk.parkingaki.beta0&hl=pt_BR. Citado na página 53.

UN, D. P. D. *2014 Revision of World Urbanization Prospects*. 2014. Disponível em: <https://esa.un.org/unpd/wup/>. Citado na página 21.

UN, D. P. D. *World's population increasingly urban with more than half living in urban areas*. 2014. Disponível em: <http://www.un.org/en/development/desa/news/population/world-urbanization-prospects-2014.html>. Citado 2 vezes nas páginas 18 e 21.

- VLAHOIANNI, E. et al. A real-time parking prediction system for smart cities. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, v. 20, n. 2, p. 192–204, 2016. Cited By 2. Citado 2 vezes nas páginas 37 e 41.
- WASSERMAN, A. I. Software engineering issues for mobile application development. In: ACM. *Proceedings of the FSE/SDP workshop on Future of software engineering research*. [S.l.], 2010. p. 397–400. Citado 3 vezes nas páginas 26, 27 e 66.
- ZHENG, Y.; RAJASEGARAR, S.; LECKIE, C. Parking availability prediction for sensor-enabled car parks in smart cities. In: *2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. [S.l.: s.n.], 2015. p. 1–6. Citado 2 vezes nas páginas 36 e 39.

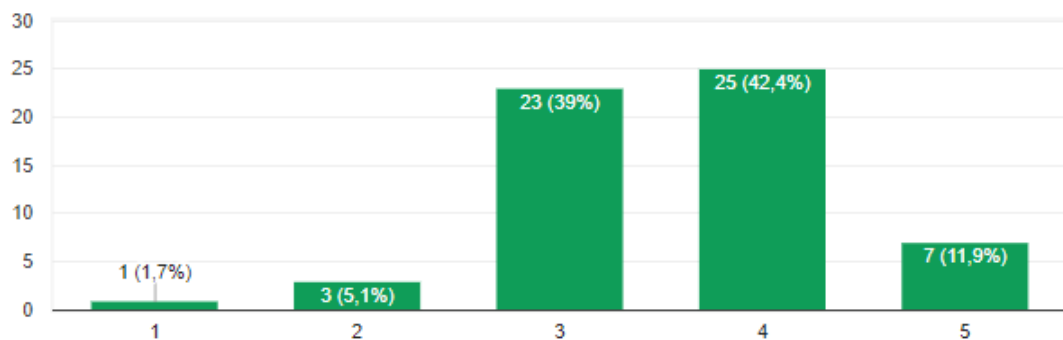
Apêndices

APÊNDICE A – Questionário com *stakeholders*

Este questionário foi utilizado para o levantamento da opinião dos possíveis utilizadores da ferramenta proposta, distribuído através do Facebook e do WhatsApp.

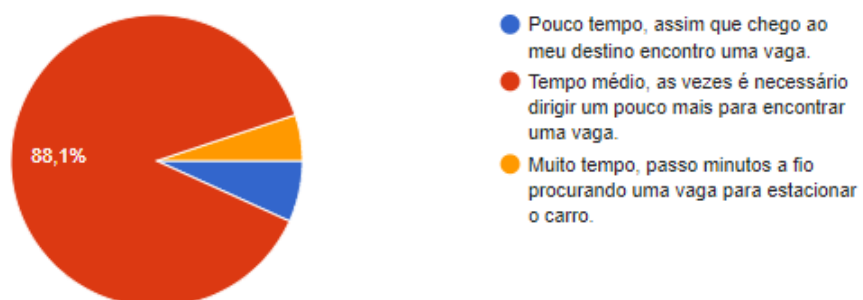
A cidade onde você reside tem problemas com congestionamento?

59 respostas



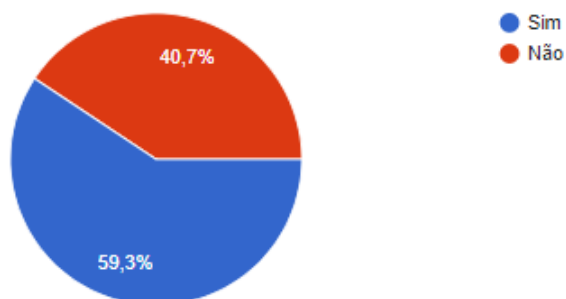
Quanto tempo em média você gasta procurando lugar para estacionar?

59 respostas



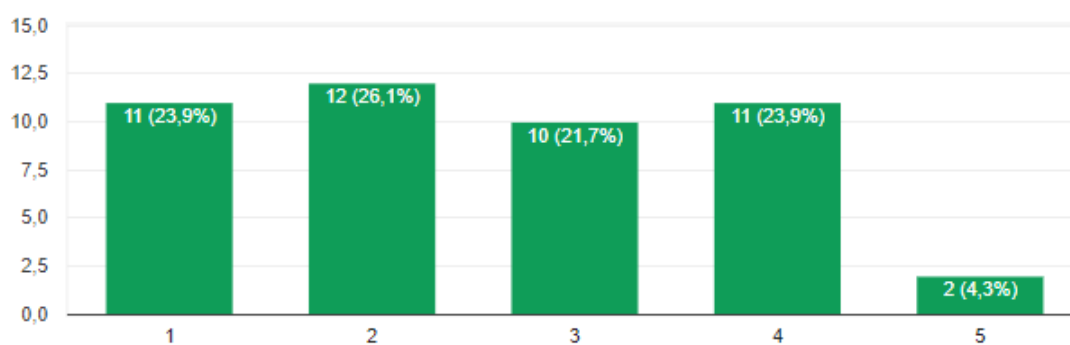
Você utiliza estacionamento privado?

59 respostas



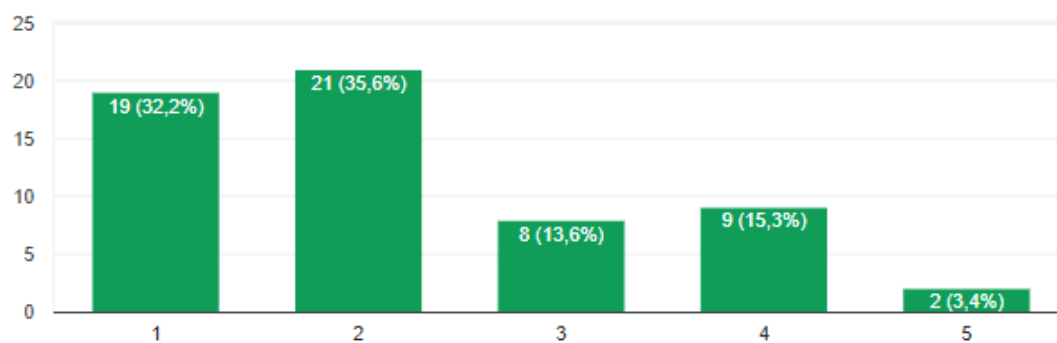
Caso sim, você utiliza o estacionamento privado com frequência?

46 respostas



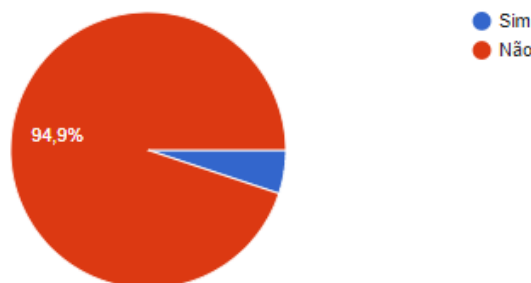
Você frequentemente esquece onde estacionou seu carro?

59 respostas



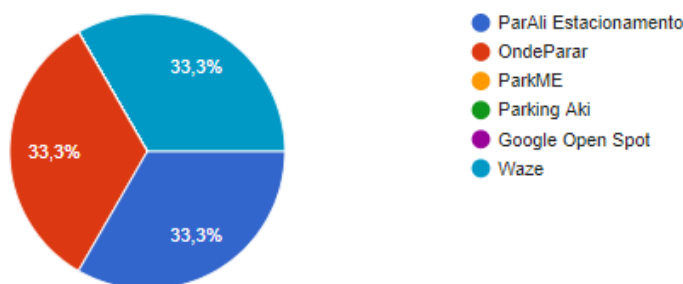
Você utiliza, ou já utilizou, algum aplicativo quando precisa encontrar vagas de estacionamento?

59 respostas



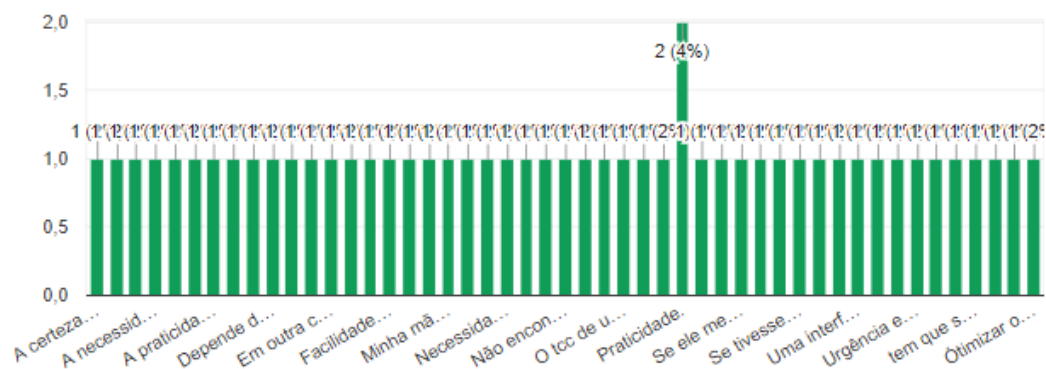
Caso sim, qual o aplicativo utilizado?

3 respostas



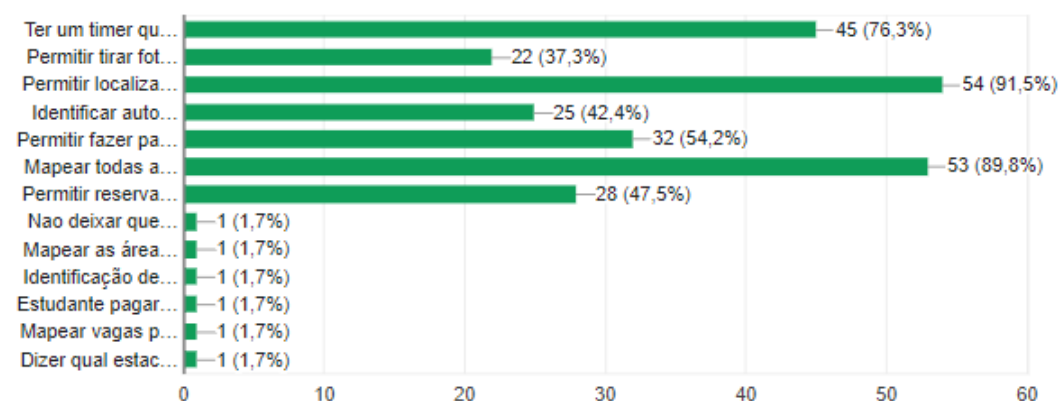
Caso não, o que te faria utilizar?

50 respostas



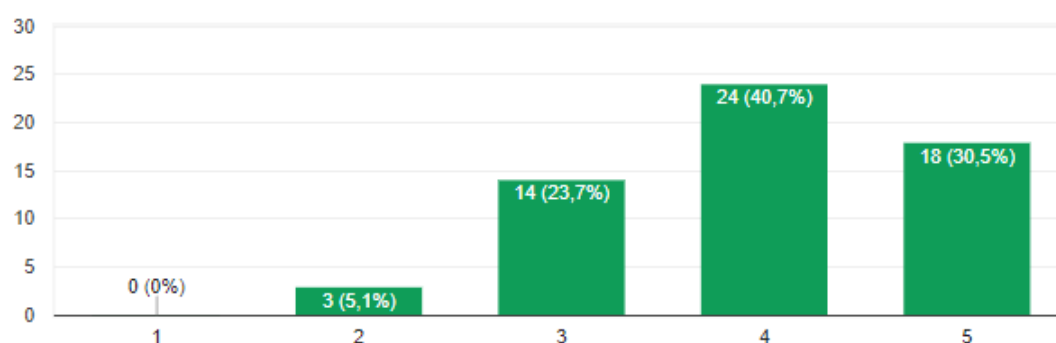
Quais características você gostaria de ter em um aplicativo que te auxiliasse a encontrar vagas para estacionar?

59 respostas



Um aplicativo desse tipo teria impacto positivo no seu dia-a-dia?

59 respostas

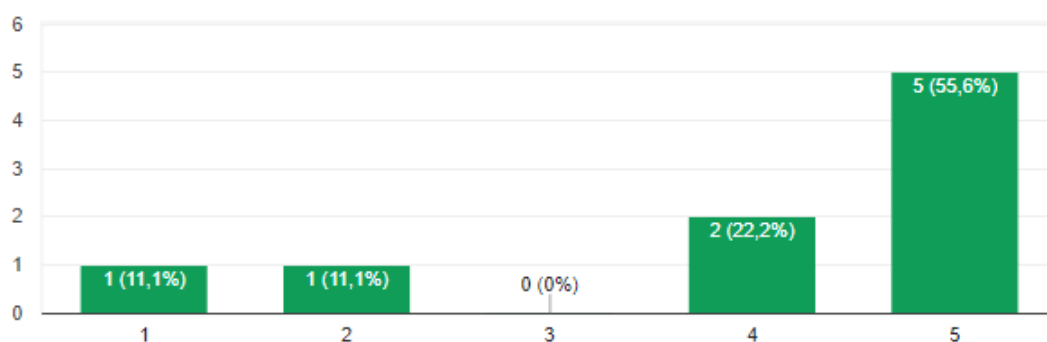


APÊNDICE B – Questionário para validação do Protótipo de Telas do Gestor de Estacionamento

Abaixo o resultado do questionário utilizado na validação do protótipo de telas voltado para o Gestor do Estacionamento, distribuído através do Facebook e do WhatsApp.

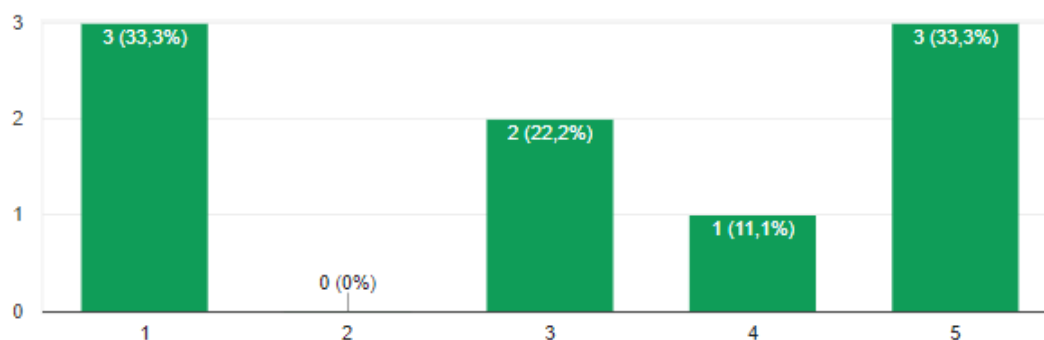
Realizar o cadastro no aplicativo lhe pareceu intuitivo?

9 respostas



Foi fácil perceber quais são os tipos de usuário que podem se cadastrar no aplicativo e quais são as opções disponíveis para eles?

9 respostas



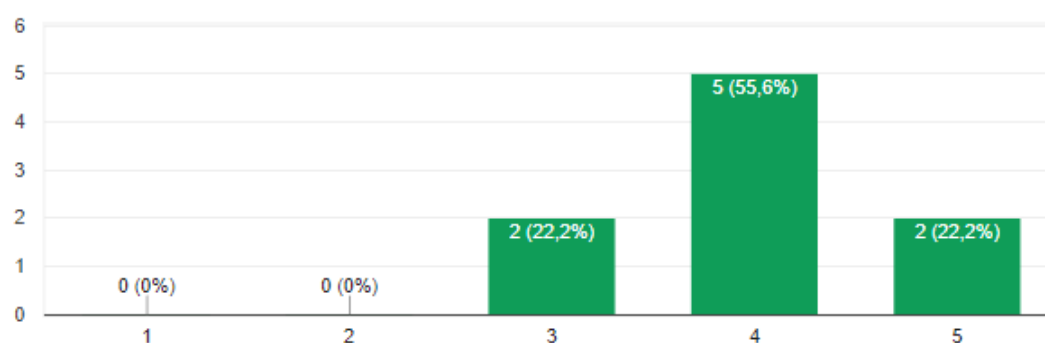
Você tem alguma sugestão do que pode ser melhorado nesta tela?

5 respostas

Não se assemelha a uma tela de cadastro, e sim de login.
Isso é a tela de login, e colocar por Gmail e Facebook seria uma boa. Uma vez que o token gerado tira a necessidade fazer um cadastro. O protótipo é imaturo, usuário final deseja ver mais "cores". Usar o pencil só se for para outro Dev ou gestor olhar.
Possibilidade de login com conta google.
Pra mim qualquer usuário pode entrar
Login com facebook ou google plus

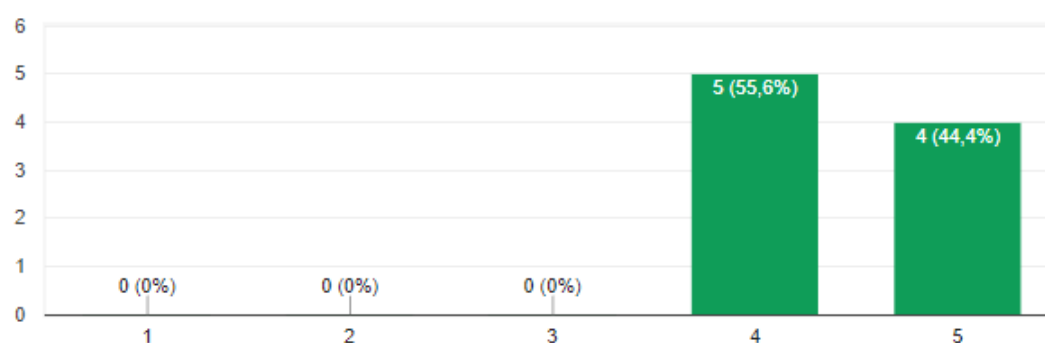
A tela inicial está intuitiva, ou seja, é fácil utiliza-la ?

9 respostas



É possível saber, através dos ícones da tela, como navegar entre as opções disponíveis?

9 respostas



Você tem alguma sugestão do que pode ser melhorado nesta tela?

2 respostas

O termo 'Gerenciar' parece redundante. Se clicamos em qualquer um dos botões, fica evidente que queremos gerenciá-lo.

Ok.

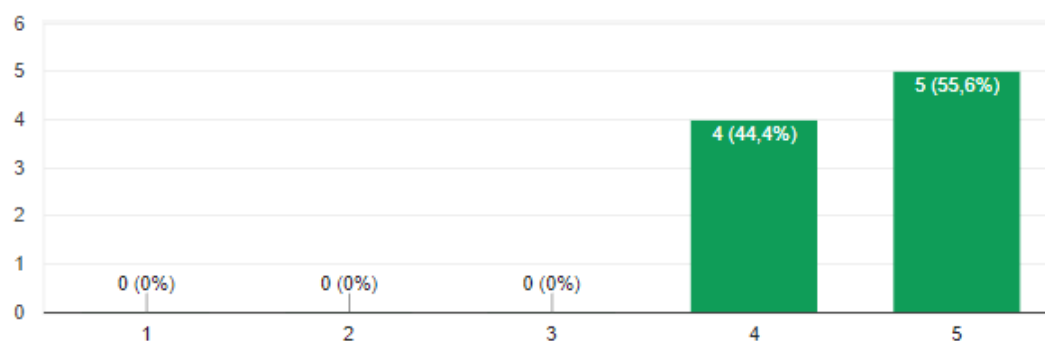
Através do menu é possível perceber e acessar as opções disponíveis ?

9 respostas



Qual a importância da tela de notificações?

9 respostas



Você tem alguma sugestão do que pode melhorar nesta tela?

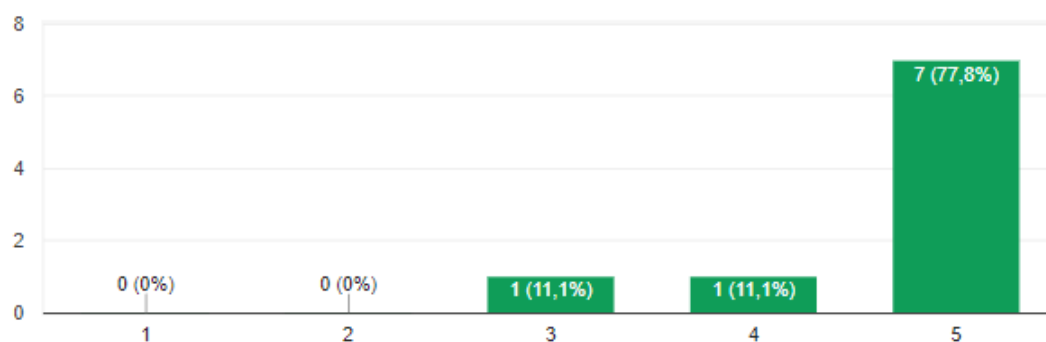
2 respostas

Um contador regressivo que indique seu prazo restante para chegar até a vaga. (se você se atrasar, perde a reserva)

Na maturação do projeto ela irá melhorar naturalmente.

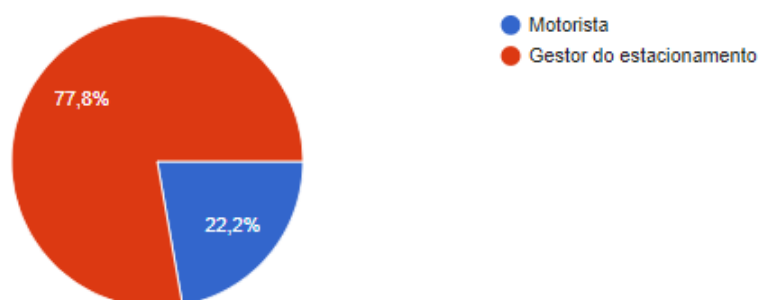
Qual a necessidade da tela com o registro da ocupação do estacionamento?

9 respostas



Na sua opinião, a opção de finalizar a ocupação da vaga deve ser do:

9 respostas



Por que?

8 respostas

Pode até ser do motorista tbm. Mas é legal que seja do gestor pq o Motorista pode esquecer.

O objetivo do app é reduzir custos aumentando a eficiência. Isso não é possível através de fiscalização, e sim de uma cultura de colaboração.

Evite depender do input do usuário. No máximo, cadastre a por coordenadas geográfica (lat. e long.) e identificar a saída da vaga por distância.

Motoristas que saiam apressados nunca iriam desempenhar este papel, o que tornaria esta função do aplicativo menos funcional.

Para disponibilidade de vagas.

Por que o usuário final não tem essa obrigação

Porque muitas pessoas esquecem, principalmente na correria do dia a dia . As pessoas podem chegar atrasadas, sair atrasadas, e esqueceriam facilmente de desabilitar a vaga, além de poderem estar sem internet ou sem bateria no momento. Pode causar transtornos a não-atualização correta do estado do estacionamento. Com um gestor, seria mais viável pois o mesmo estaria provido dos materiais necessários para melhor gerir o estacionamento, além de estar mais atento a quem entra e quem sai do estacionamento, permitindo um melhor controle.

Ele é quem vai levar o carro e precisa saber antes se têm vaga

Você tem alguma sugestão do que pode ser melhorado nesta tela?

2 respostas

Apesar de motorista de má fé poder encerrar a ocupação sem de fato liberar a vaga, o motorista seguinte (que reservou o horário) deve poder denunciar o abuso , preferencialmente por dentro do próprio app e com envio de fotografia.

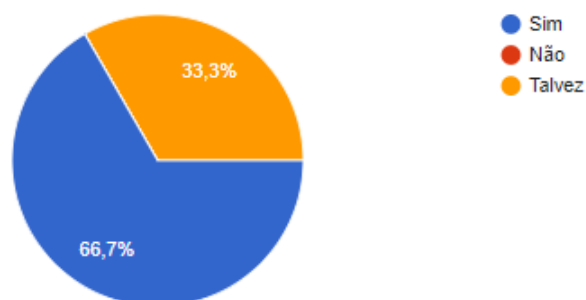
A placa servirá para identificar o infrator.

A opção de finalizar a ocupação da vaga talvez pudesse estar disponível para motoristas e para o gestor do estacionamento.

Tela de Gerencia de Estacionamentos

A possibilidade de cadastrar múltiplos estacionamentos é interessante para o usuário?

9 respostas



Em caso não positivo, explique o porquê.

0 resposta

Ainda não há respostas para esta pergunta.

Em caso não positivo, explique o porquê.

0 resposta

Ainda não há respostas para esta pergunta.

Você tem alguma sugestão do que pode ser melhorado nesta tela?

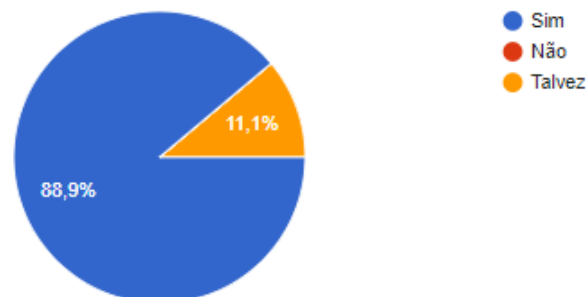
0 resposta

Ainda não há respostas para esta pergunta.

Tela de alterar perfil

A tela de alterar perfil possui opções suficientes?

9 respostas



Em caso não positivo, explique o porquê.

0 resposta

Ainda não há respostas para esta pergunta.

Você tem alguma sugestão do que pode ser melhorado nesta tela?

3 respostas

Talvez o registro de e-mail possa ser substituído por registro de número telefônico, para tokens SMS. Enquanto o cliente está na rua, procurando vagas, o número telefônico geralmente está mais acessível (está no próprio aparelho) que uma conta de e-mail (que nem sempre está configurada no smartphone do cliente).

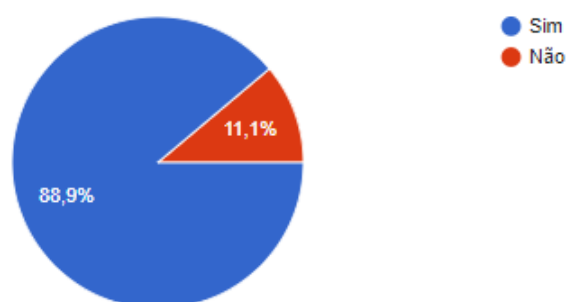
Aí vai do estudo exploratório sobre a necessidade do gestor dos estacionamento.

Envie por e-mail informações de que vaga o carro foi colocado. Um link do maps talvez.

Tela de cadastro do estacionamento

Os campos apresentados no formulários são suficientes para o cadastro do estacionamento ?

9 respostas



Se não, qual novo campo deve ser inserido?

3 respostas

O gestor é quem decide boa parte.

Campos necessários para tratamento das vagas especiais.

Qualidade do estacionamento. Pode ser colocado com estrelas pra avaliar o atendimento

Você tem alguma sugestão do que pode ser melhorado nesta tela?

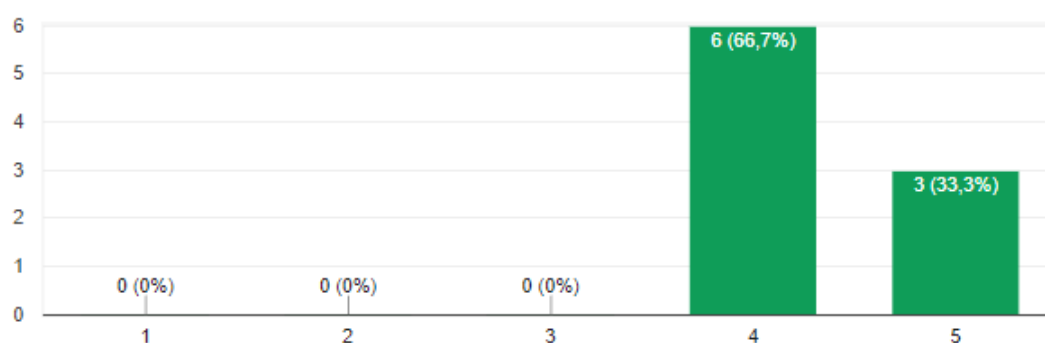
1 resposta

Alguns estabelecimentos tem horários distintos entre dias diferentes, como sábados. É necessário dar essa opção.

Tela de relatórios

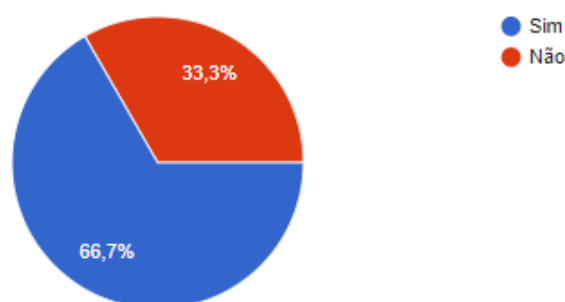
Qual a importância da geração de relatórios para um aplicativo deste tipo?

9 respostas



As opções de relatório são suficientes para um gestor de estacionamento?

9 respostas



Se não, por quê?

3 respostas

Na vdd depende das especificações de cada. Das possibilidades de verificar qnd o estacionamento ficou mais ocupado, reservas canceladas ..essas coisas

Não sou gestor.

Dependendo do estacionamento outros relatórios poderiam ser necessários, como por exemplo, horários de pico e dias especiais.

Você tem alguma sugestão do que pode ser melhorado nesta tela?

1 resposta

Deveria ter a opção 'talvez' ou não opinar nessa questão.

Deixe sua opinião!

De forma geral qual a sua opinião em relação ao protótipo do aplicativo?
Você tem alguma sugestão que poderia melhorar o aplicativo?

6 respostas

Tá bem legal. Parabéns

Bem prematuro, mas é assim mesmo. Já ganham free feedback e tal. Dica, como o TCC é tecnológico corte etapas desnecessárias da gestão, uma vez que algum delas são somente papel e iram retardar o desenvolvimento. Mas caso a equipe seja grande, muita delas são realmente essenciais. Porém, muitas delas referentes ao século passado.

Muito bom.

Excelente! Fácil uso e facilitaria muito a vida do motorista usuário.

Sistema de avaliação do estacionamento por estrelas que indicam qualidade. Informações se as vagas são cobertas ou não. Probabilidade de perda ou roubo do veículo nas proximidades do estacionamento que pode ser informado pelos usuários. Notificar o usuário por e-mail a localização do seu veículo quando está na vaga e quando a deixa.

Achei muito bom

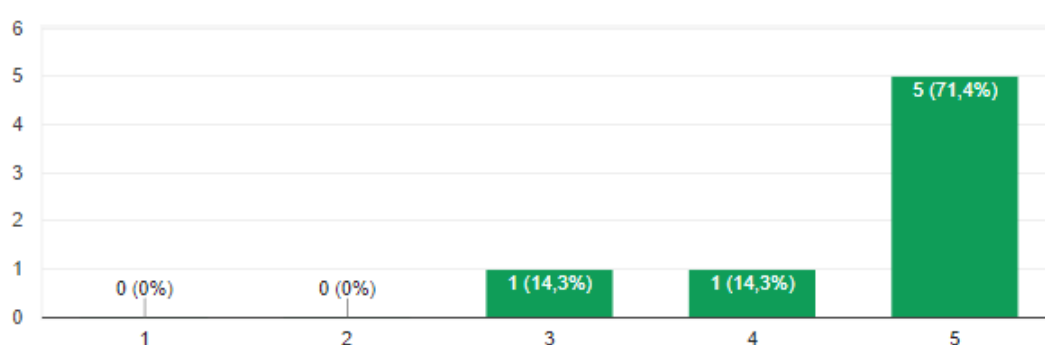
APÊNDICE C – Questionário para validação dos Protótipo de Telas do Motorista

Abaixo o resultado do questionário utilizado na validação do protótipo de telas voltado para o Motorista, distribuído através do Facebook e do WhatsApp.

Autenticação no aplicativo

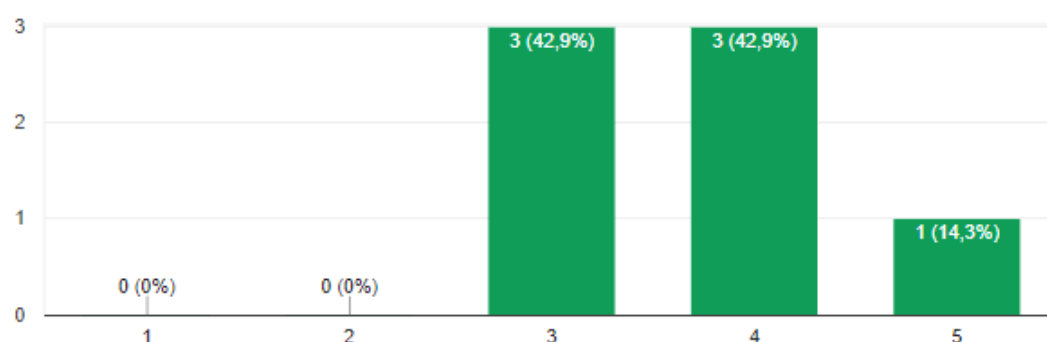
Realizar o cadastro no aplicativo lhe pareceu intuitivo?

7 respostas



Foi fácil perceber quais são os tipos de usuário que podem se cadastrar no aplicativo e quais são as opções disponíveis para eles?

7 respostas



Você tem alguma sugestão do que pode ser melhorado nesta tela?

3 respostas

Login através da conta Google.

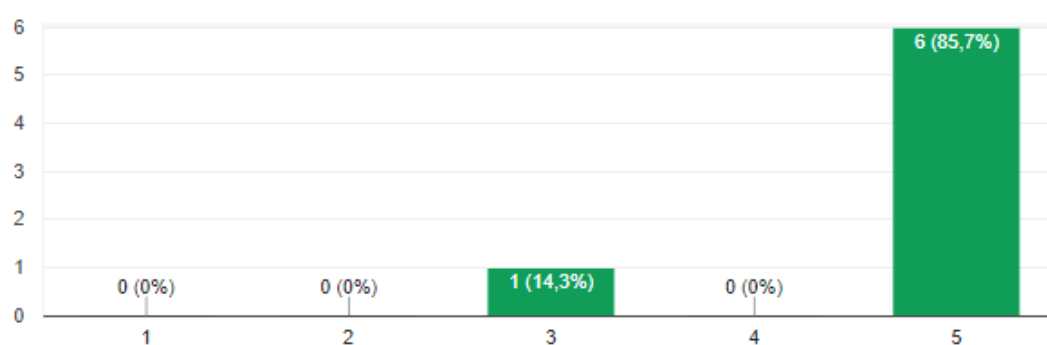
Seria bom um texto abaixo fazendo um breve explicação sobre o App.

Se os tipos de usuário são apenas dois e não irão mudar, talvez seja melhor um radio group pois o usuário já veria todas as opções sem precisar clicar no combobox.

Tela Inicial

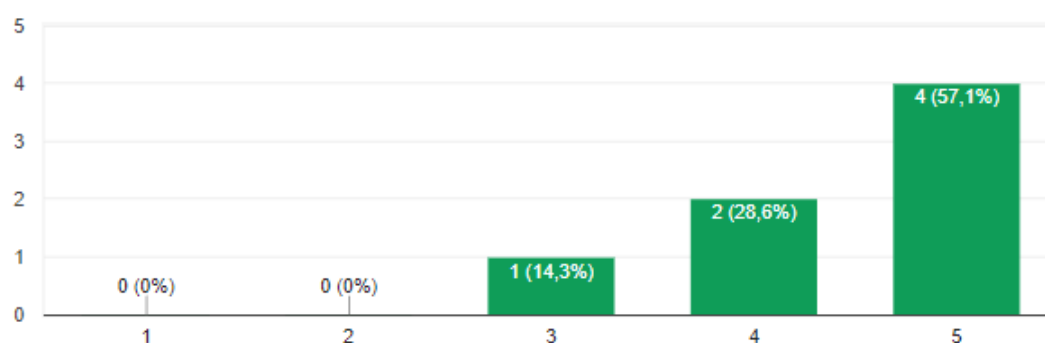
A tela inicial está intuitiva, ou seja, é fácil utiliza-la ?

7 respostas



É possível saber, através dos ícones da tela, como navegar entre as opções disponíveis?

7 respostas



Você tem alguma sugestão do que pode ser melhorado nesta tela?

2 respostas

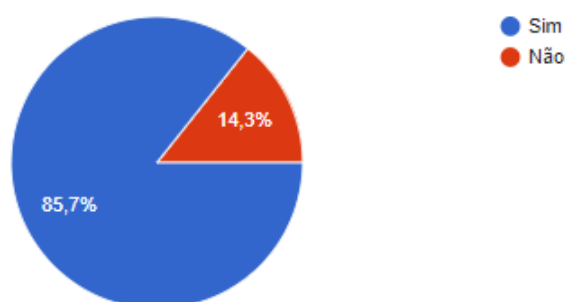
Acho que faltou um título. (Uma opção de Logout em algum momento eu não vi)

Os ícones de localização dos estacionamentos poderiam usar cores diferentes indicando se possuem muitas vagas livres ou não.

Tela de vagas

A tela de vagas remete o seu propósito?

7 respostas



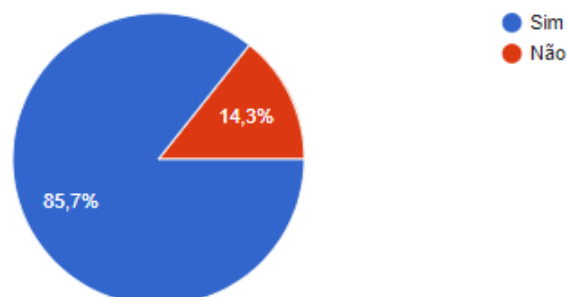
Se não, por quê?

0 resposta

Ainda não há respostas para esta pergunta.

As cores usadas são adequadas para representar o status de cada vaga?

7 respostas



Se não, por quê?

2 respostas

Não consegui identificar o que significa a cor amarela.

Acho que para vagas especiais e reservadas deveria ser mais clara

Em caso não positivo, explique o porquê.

2 respostas

Deixaria de disponibilizar a vaga para alguém que foi até o local de forma não planejada, podendo haver a possibilidade de quem reservou não usufruir de fato.

Tem que deixar claro o tempo de disponibilidade e economia, pois tem muitos estacionamento que tem uma grande ociosidade. Talvez outros não achem interessante por ter alta procura.

Você tem alguma sugestão do que pode melhorar nesta tela?

3 respostas

A tela de vagas poderia ter legendas embaixo explicando o significado de cada cor.

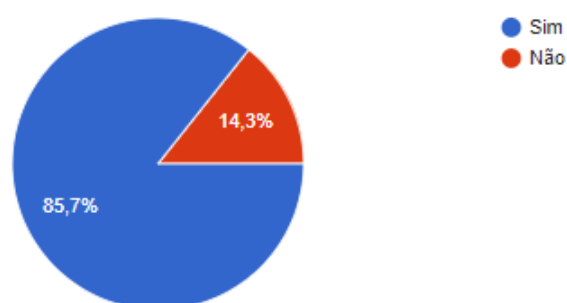
Acho que tem que ser uma tela resumida informando a quantidade tipos de vagas disponíveis. E assim o pessoal faz a reserva

Talvez não seja claro pra todos os usuários a diferença entre as vagas na cor vermelha e amarela. Poderia ter uma pequena legenda na parte inferior da tela.

Menu de Opções

Através do menu é possível perceber e acessar as opções disponíveis ?

7 respostas



Você tem alguma sugestão do que pode melhorar nesta tela?

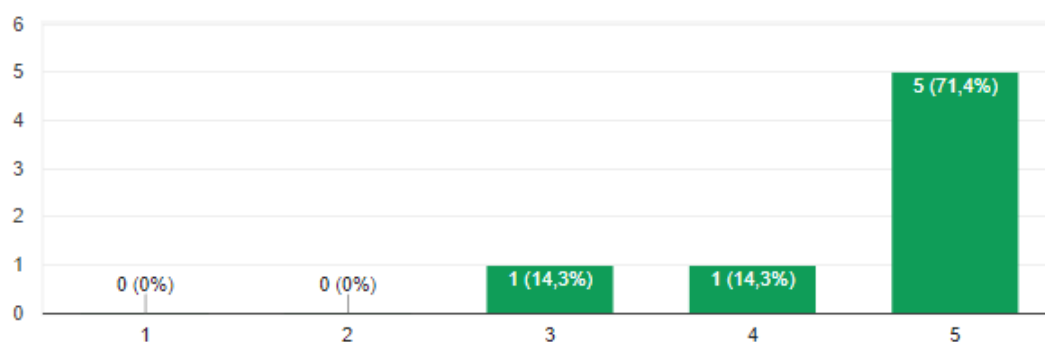
1 resposta

Acho que falta um título

Tela de Notificações

Qual a importância da tela de notificações?

7 respostas



Você tem alguma sugestão do que pode melhorar nesta tela?

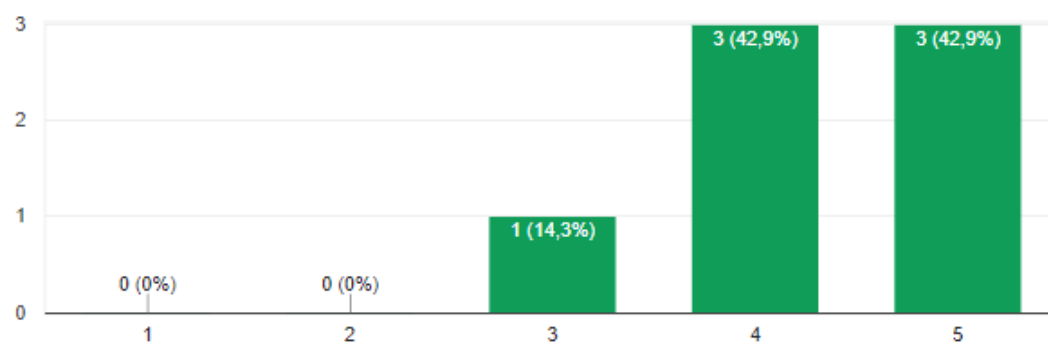
1 resposta

Possibilidade de configurar notificações. Por ex, configurar que o app alerte quando faltar X minutos (ao invés de tempo fixo) pois dependendo da situação o usuário pode precisar se dirigir ao carro com maior antecedência.

Tela de Localizar Carro

Qual a importância de uma opção para localizar o seu carro estacionado?

7 respostas



Você tem alguma sugestão do que pode melhorar nesta tela?

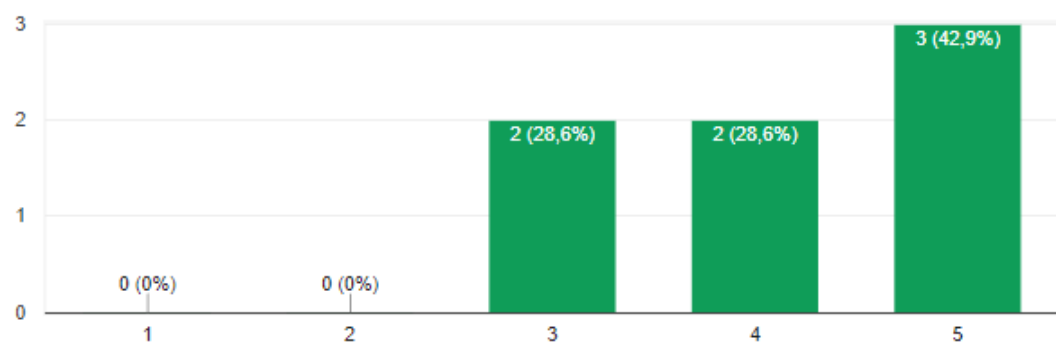
1 resposta

O ícone do usuário poderia incluir uma seta indicando a direção em que ele está se movendo (útil quando a atualização da posição do ícone de usuário levar algum tempo)

Tela de histórico

Qual a necessidade da tela de histórico?

7 respostas



Você tem alguma sugestão do que pode melhorar nesta tela?

2 respostas

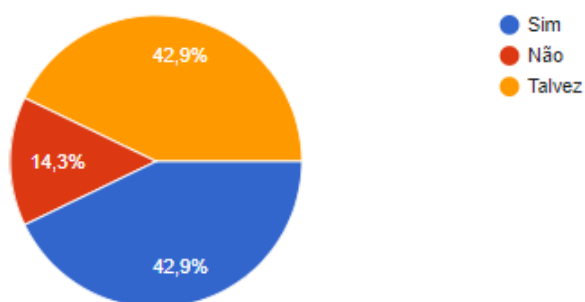
É interessante pois a pessoa pode ir ao mesmo estacionamento ou não.

A ordenação poderia ser do mais recente pro mais antigo, ou ter a opção de mudar a ordenação clicando no cabeçalho de Data (que teria uma seta pra cima ou pra baixo de acordo com a ordenação atual).

Tela de Gerencia de Veículos

A possibilidade de cadastrar múltiplos veículos é interessante para o usuário?

7 respostas



Em caso não positivo, explique o porquê.

1 resposta

Acho que pode ser uma opção para a pessoa se organizar melhor.

Você tem alguma sugestão do que pode melhorar nesta tela?

0 resposta

Ainda não há respostas para esta pergunta.

Você tem alguma sugestão do que pode melhorar nesta tela?

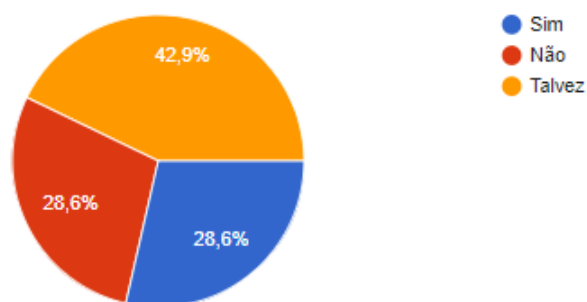
0 resposta

Ainda não há respostas para esta pergunta.

Tela de alterar perfil

A tela de alterar perfil possui opções suficientes?

7 respostas



Em caso não positivo, explique o porquê.

3 respostas

Falta dados do carro

A parte do cartão de credito creio que tem que deixar mais para os ecomerce dos gerenciadores de cartão para questão de segurança. Creio que ter uma opção da pessoa descontar credits de algum tipo de cartão pre pago,

Talvez seja interessante informar também o número de telefone pro caso dos funcionários do estacionamento precisarem entrar em contato com o motorista.

Você tem alguma sugestão do que pode melhorar nesta tela?

2 respostas

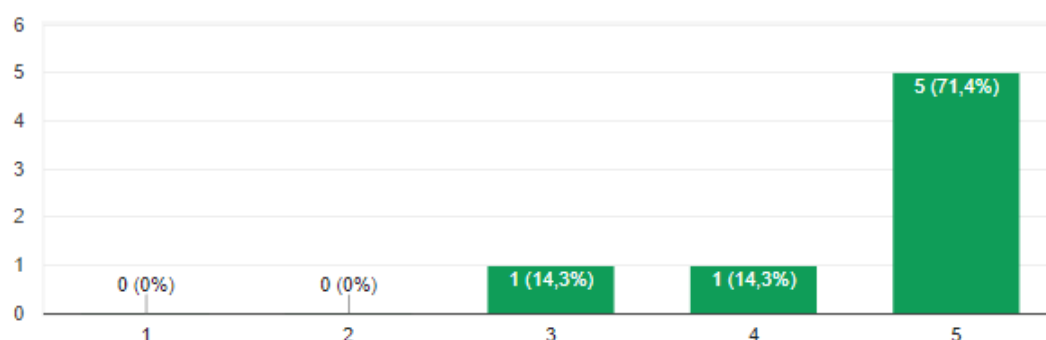
Não sei se será o caso, mas lembrar de deixar somente os 4 últimos e 4 primeiros dígitos do cartão, não sei qual o padrão utilizado, em exibição, com o restante do número em asteriscos como questão de segurança.

Avaliar a questão dos dados dos cartões, sobre a segurança

Tela de formas de pagamento

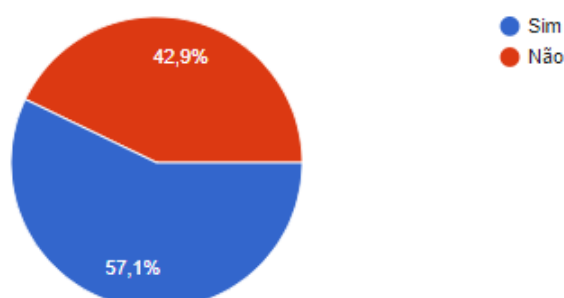
Qual a importância de pagamentos com cartão de crédito em um aplicativo como esse?

7 respostas



Os campos apresentados no formulário são suficientes para o cadastro adequado do meio de pagamento?

7 respostas



Se não, por que?

2 respostas

falta cpf e nascimento do dono

Acho que tem muita informação

Você tem alguma sugestão do que pode melhorar nesta tela?

2 respostas

Dar opção de cartão pre pago.

Colocar uma imagem ao lado do CVV indicando onde encontrar essa informação no cartão.

Deixe sua opinião!

De forma geral qual a sua opinião em relação ao protótipo do aplicativo? Você tem alguma sugestão que poderia melhorar o aplicativo?

5 respostas

Muito legal.. Principalmente a parte de ver onde você estacionou é o de você esta

Acredito que está simples e direto, muito útil! Acredito que seria legal uma versão "pública", sem a necessidade de cadastro de informações de pagamento e com a disponibilização de buscar vagas somente quando estiver próximo do local para lugares públicos, como a UFS por exemplo.

Muito bom.

Poucas melhorias nas telas como títulos das opções.

Gostei, é fácil de entender e as funções podem ser acessadas rapidamente.

APÊNDICE D – Questionário para validação do aplicativo desenvolvido - Usuário Gestor de Estacionamento

Abaixo o resultado do questionário utilizado na validação do aplicativo desenvolvido voltado para usuários do tipo Gestor de Estacionamento, realizado através de visitas em estacionamentos da cidade de Aracaju.

Perfil do respondente

4 respostas

Trabalha em um estacionamento no centro de Aracaju há um ano e oito meses. Gerencia entradas e saídas de veículos.

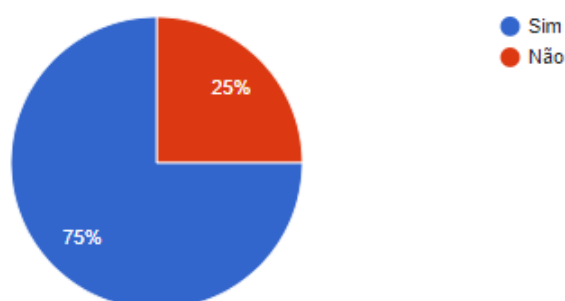
Trabalha com estacionamento há um ano e meio. Verificando entrada e saída de veículo

Trabalho há 4 anos com estacionamento. Verificando e registrando entradas e saídas de veículos

Trabalha há 7 anos com estacionamento no centro de Aracaju. Registrando entradas e saídas de vagas.

Você utiliza alguma ferramenta/sistema para gerenciar seu estacionamento?

4 respostas



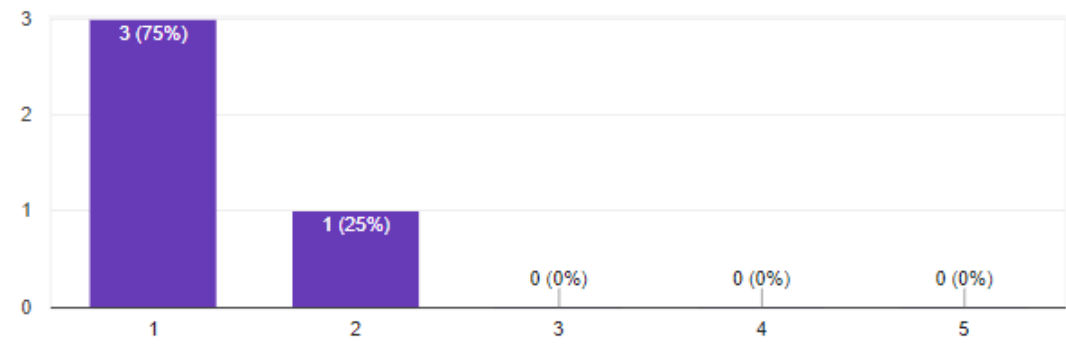
Qual a ferramenta/sistema que você utiliza?

3 respostas

CopyPaste
Maquina registradora
Programa século

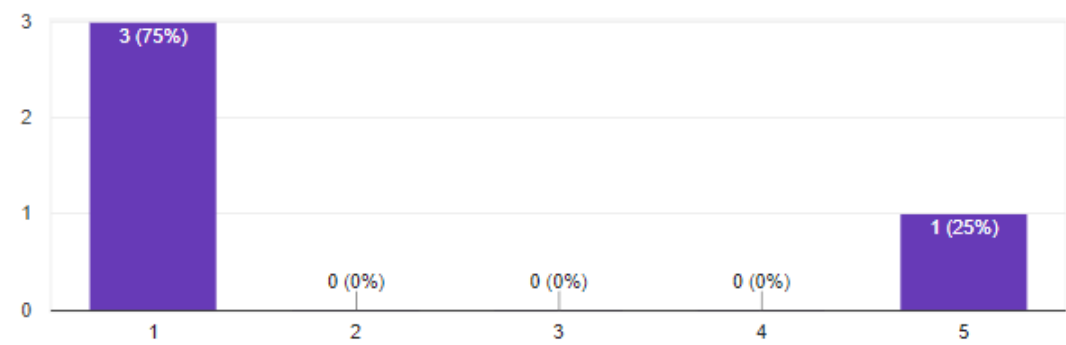
Qual o nível de dificuldade no momento de cadastrar um usuário?

4 respostas



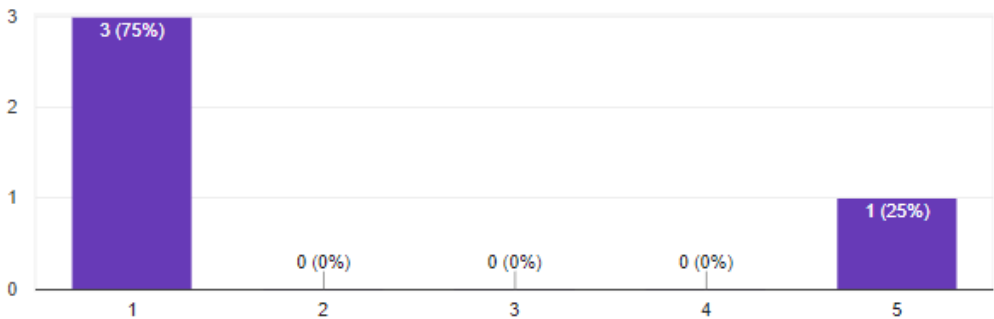
Qual o nível de dificuldade no momento de cadastrar um estacionamento?

4 respostas



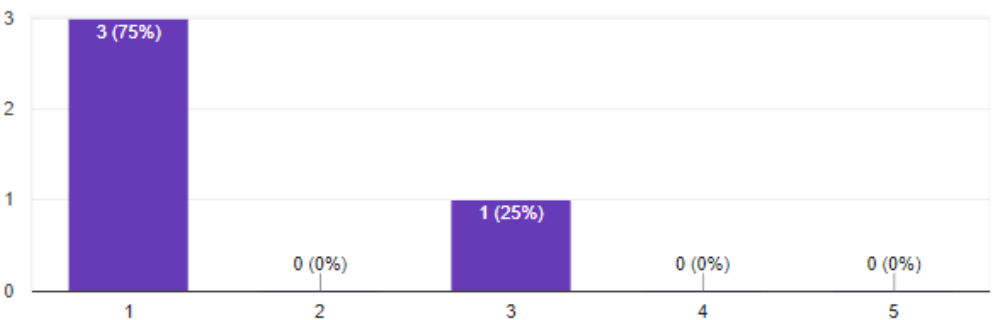
Qual o nível de dificuldade na atualização do perfil?

4 respostas



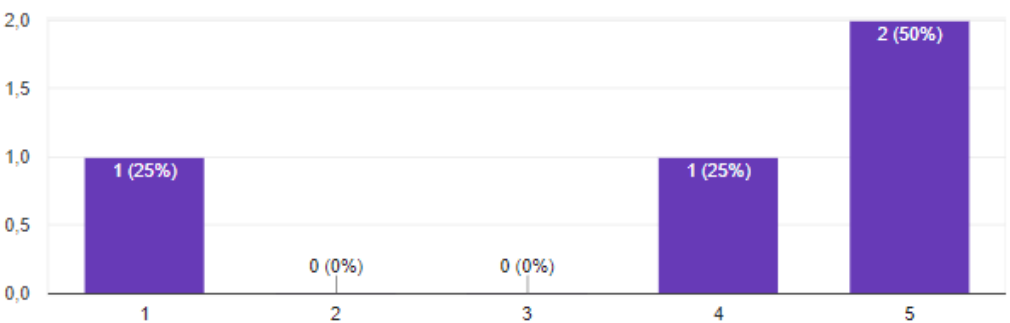
Qual o nível de dificuldade na recuperação da senha?

4 respostas



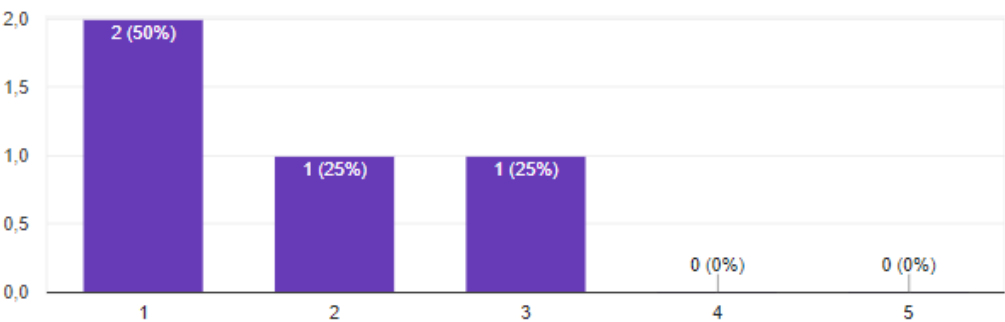
A interface deixa claro a funcionalidade de cada opção?

4 respostas



De forma geral qual a dificuldade sentida durante o uso da aplicação?

4 respostas



Na sua opinião, quais os pontos fracos do aplicativo?

3 respostas

Precisar atualizar o status da vaga

As reservas não são interessantes devido a rotatividade.

A reserva não é interessante devido a rotatividade. Um app apresentou um erro ao clicar no estacionamento cadastrado.

Na sua opinião, quais os pontos fortes do aplicativo?

4 respostas

A facilidade de encontrar o estacionamento no mapa e ter acesso a disponibilidade da vaga

A questão da divulgação, o fato de o estacionamento aparecer no mapa pra o usuário vê.

O fato de informar se tem vaga ou não.

É um aplicativo interessante para ser usado aqui em Aracaju. Existe muitas necessidades nessa área.

Você utilizaria o SmartParking para gerenciar seu estacionamento?

4 respostas



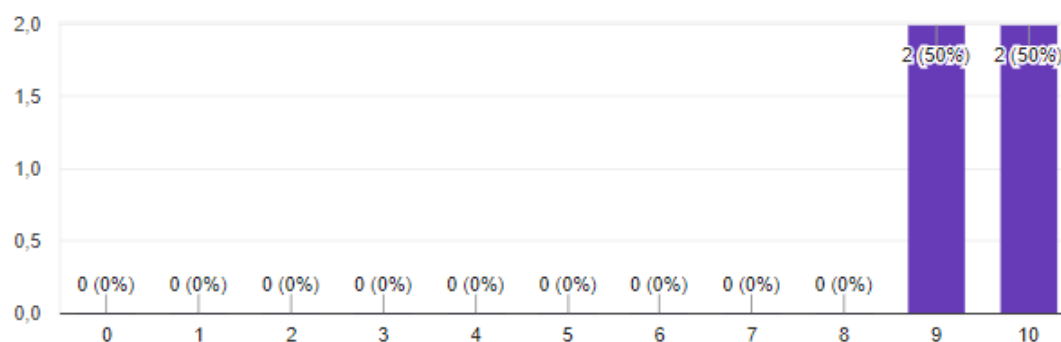
Quais as melhorias que você sugere?

4 respostas

Inserir alguma ajuda nas opções do aplicativo
Seria interessante mandar uma notificação para confirmar se o estacionamento tem vaga ou não. Às vezes na correria a pessoa pode esquecer de atualizar o status da vaga. Divulgar serviços de lavagem que estacionamento faz.
Nenhuma
Ter a possibilidade de reservar o serviço pelo aplicativo, como lavagem. Existe muita procura.

Pensando nas funcionalidades que este aplicativo propõe, de 0 a 10, qual a nota que você atribuiria ao aplicativo?

4 respostas



APÊNDICE E – Questionário para validação do aplicativo desenvolvido - Usuário Motorista

Abaixo o resultado do questionário utilizado na validação do aplicativo desenvolvido voltado para usuários do tipo Motorista, realizado com motoristas experientes da cidade de Aracaju.

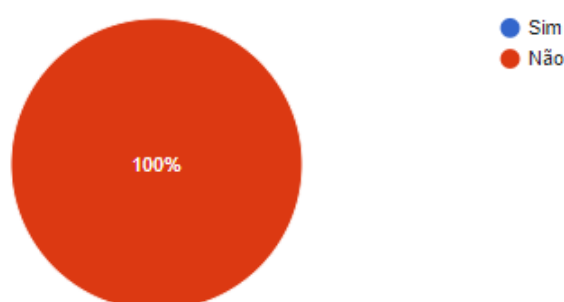
Perfil do respondente

5 respostas

Motorista (2)
Ícaro
Motorista com 10 anos de experiência
Motorista a 4 anos

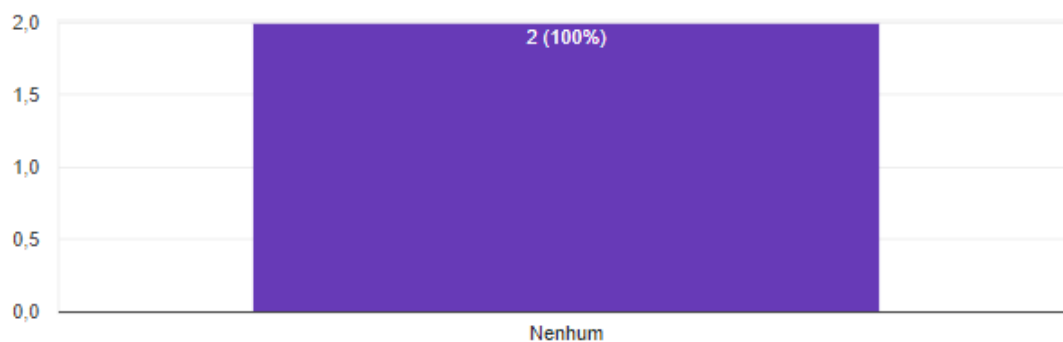
Você já utiliza algum aplicativo para buscar vagas de estacionamento?

6 respostas



Qual aplicativo você utiliza?

2 respostas



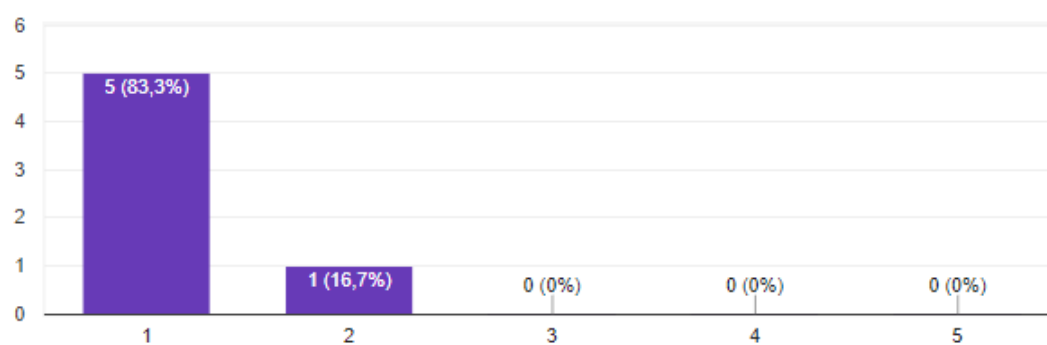
Você usaria o Smart Parking para procurar vagas para estacionar?

6 respostas



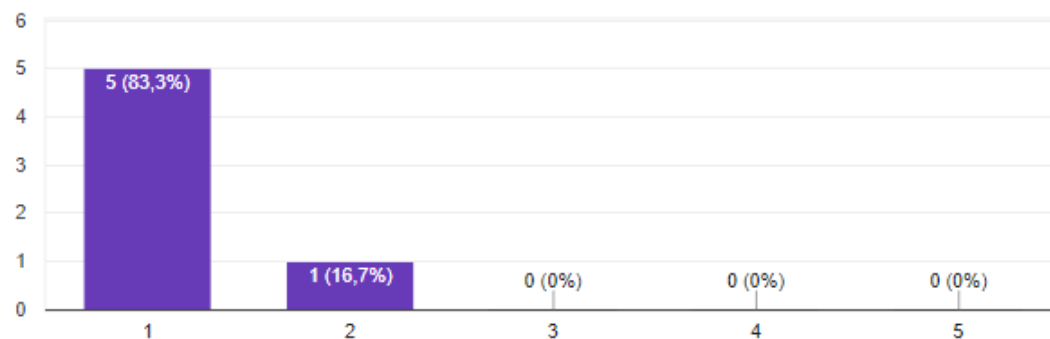
Qual nível de dificuldade no cadastro do usuário?

6 respostas



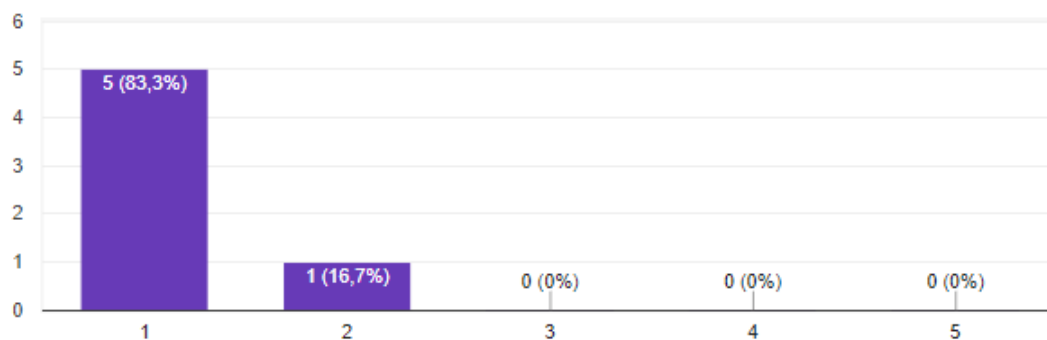
Qual o nível de dificuldade no gerenciamento dos veículos?

6 respostas



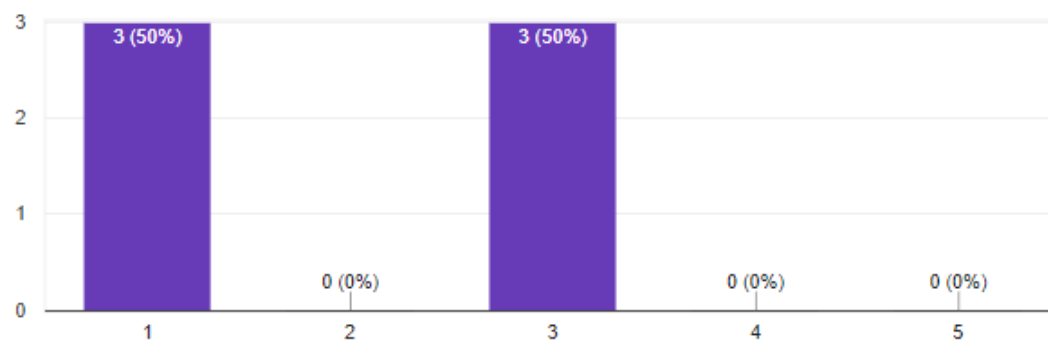
Qual o nível de dificuldade na atualização do perfil?

6 respostas



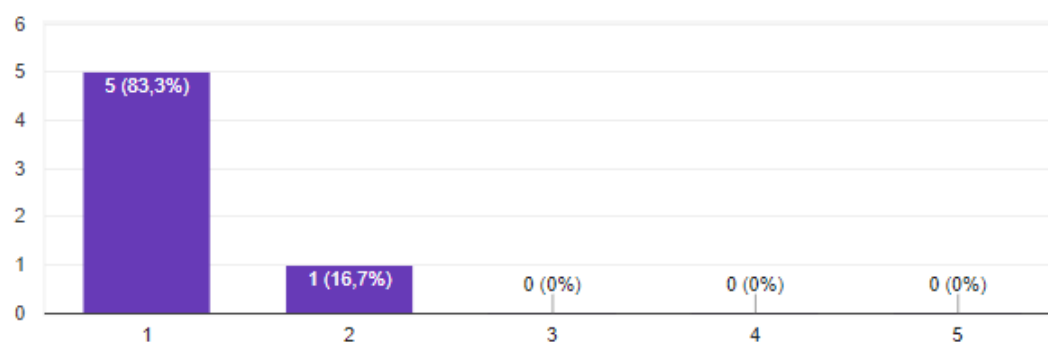
Qual o nível de dificuldade na busca por vagas?

6 respostas



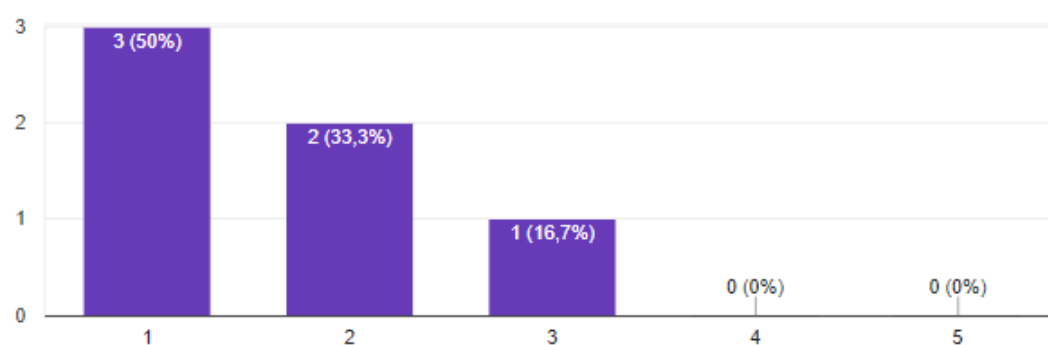
Qual o nível de dificuldade na recuperação de senha?

6 respostas



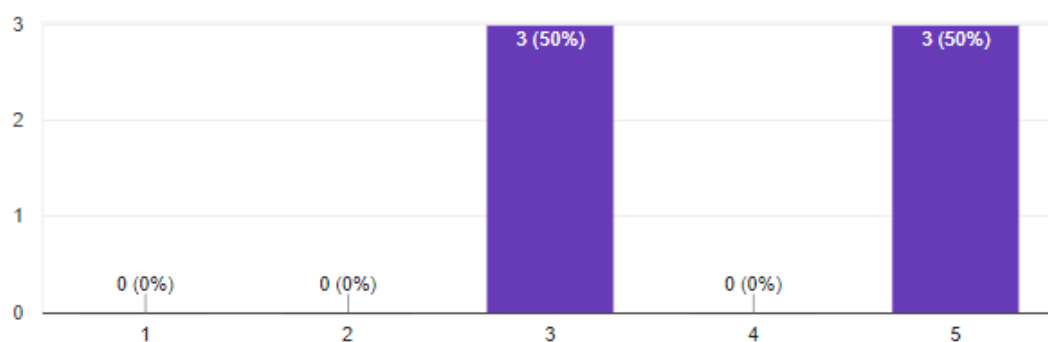
Qual o nível de dificuldade no uso do aplicativo em geral?

6 respostas



A interface deixa clara a funcionalidade de cada opção?

6 respostas



Na sua opinião, quais os pontos fortes do aplicativo?

5 respostas

O mapa e o gps.
Simplicidade, fluido, organizado
Ficou fácil encontrar o estacionamento que eu queria.
Poder reservar uma vaga de estacionamento.
A facilidade de encontrar vagas disponíveis na região

Na sua opinião, quais são os pontos fracos do aplicativo?

5 respostas

Nenhum (2)
Travamento.
Cores da pesquisa ficam um pouco difícil de enxergar,
Não ter uma lista maior de vagas de estacionamento.

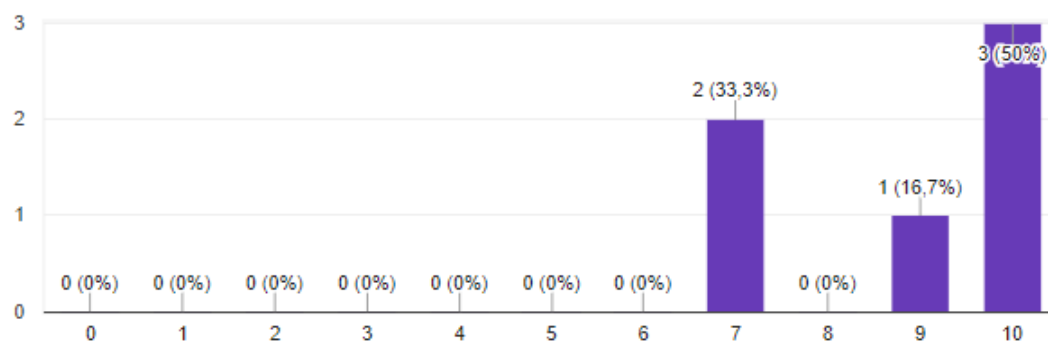
Quais melhorias você sugere?

5 respostas

Há atualizações a serem feitas.
No gerenciador de veículo colocar opções de gerenciando de troca de óleo, troca para manutenção preventiva.
Nenhum
Ser mais intuitivo, ter uma passo a passo.
Tá tudo ótimo

Pensando nas funcionalidades que este aplicativo propõe, de 0 a 10, qual a nota que você atribuiria ao aplicativo?

6 respostas



APÊNDICE F – Projeto de Testes Smart Parking

Abaixo segue o projeto de testes para o Smart Parking, o qual descreve os casos e procedimentos de testes, assim como os resultados da aplicação destes pelos testadores envolvidos.

Projeto de Testes *Smart Parking*

Versão 1.0

**Fase de Desenvolvimento: Teste de Caixa
Preta**

Interação 01

Sumário

PROJETO DE TESTES SMART PARKING.....	1
VERSÃO 1.0	1
FASE DE DESENVOLVIMENTO: TESTE DE CAIXA PRETA	1
INTERAÇÃO 01	1
1. INTRODUÇÃO	3
2. CASOS DE TESTE	3
2.1 USUÁRIO MOTORISTA	3
2.2 USUÁRIO GESTOR DO ESTACIONAMENTO	6
3. PROCEDIMENTOS DE TESTE	9
3.1 USUÁRIO MOTORISTA	9
3.2 USUÁRIO GESTOR DO ESTACIONAMENTO	12
4. SUMÁRIO DE AVALIAÇÃO DE TESTES – TESTADOR 1	16
4.1 USUÁRIO MOTORISTA	16
4.2 USUÁRIO GESTOR DO ESTACIONAMENTO	19
5. SUMÁRIO DE AVALIAÇÃO DE TESTES – TESTADOR 2	22
5.1 USUÁRIO MOTORISTA	22
5.2 USUÁRIO GESTOR DO ESTACIONAMENTO	25
6. SUMÁRIO DE AVALIAÇÃO DE TESTES – TESTADOR 3	29
6.1 USUÁRIO MOTORISTA	29
6.2 USUÁRIO GESTOR DO ESTACIONAMENTO	32

1. Introdução

Este documento descreve os casos e procedimentos de testes para execução no aplicativo Smart Parking, assim como o resultado dos testes de caixa preta realizados por três alunos do curso de ciência da computação da Universidade Federal de Sergipe.

Endereço do repositório:

<https://SamilaRuane@bitbucket.org/SamilaRuane/smartparking.git>

2. Casos de Teste

Esta seção descreve os casos de testes para cada funcionalidade do aplicativo separados por tipo de usuário.

2.1 Usuário Motorista

[CT001] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo motorista.	
Requisito(s) Associado (s): RF01	
Pré Condições: <ul style="list-style-type: none">• Possuir o aplicativo instalado no smartphone utilizado.• Estar conectado à rede Wifi	Pós Condições: <ul style="list-style-type: none">• O usuário deve ser cadastrado no sistema.• As credenciais devem ser cadastradas no <i>Firebase</i>.• O novo nó com o registro do usuário deve ser salvo no ramo de Usuários no <i>Realtime Database</i>.
Critérios de Sucesso: <ul style="list-style-type: none">• Pós condições satisfeitas• Uma mensagem de confirmação deve ser exibida caso as informações inseridas estejam corretas.	
Procedimento Associado: [PT001]	

[CT002] Autenticar-se no Sistema	
Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu email e senha cadastrados	
Requisito(s) Associado(s): RF01	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve acessar a página inicial do aplicativo e ter acesso a todas as opções que o app oferece de acordo com o tipo de usuário do cadastro.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • Um <i>toast</i> informando “Operação realizada com sucesso” deve aparecer na tela. 	
Procedimento Associado: [PT002]	

[CT003] Atualizar informações do Cadastro	
Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.	
Requisito(s) Associado(s): RF01	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve ter acesso a página de edição com suas informações previamente preenchidas de acordo com o que foi informado no cadastro inicial.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • Um <i>Toast</i> informando “Operação realizada com sucesso” deve aparecer na tela. 	
Procedimento Associado: [PT003]	

[CT004] Cadastrar veículos	
Descrição: O responsável pelo teste deve tentar cadastrar um veículo no sistema.	
Requisito(s) Associado(s): RF10	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve ter acesso a uma página principal com as informações do veículo definido como atual e quando necessário editar as informações do veículo assim como adicionar novos veículos que podem ser definidos como atual quando necessário. Também deve ser possível excluir e editar um veículo.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • A cada novo veículo cadastrado, este aparece na lista de veículos cadastrados. • Ao clicar em um carro da lista as informações deste deve aparecer na página principal como o carro atual escolhido. 	
Procedimento Associado: [PT004]	

[CT005] Procurar vagas para estacionar	
Descrição: O responsável pelo teste deve informar um local de destino e o sistema irá retornar todos os estacionamentos próximos com vagas em um raio de 1 Km.	
Requisito(s) Associado(s): RF11	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar com o GPS ativado e conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • Depois de inseridas as informações do destino o sistema deve retornar todos os estacionamentos existentes a uma distância de 1Km.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • O usuário deve conseguir clicar no ícone do estacionamento desejado e ao escolher a opção “vagas”, este deve ser redirecionado para a página de vagas. • Ao escolher um estacionamento e escolher a opção “rota” as informações da rota da origem do usuário até o seu destino devem ser exibidas para direcionar o usuário até aquele estacionamento. 	
Procedimento Associado: [PT005]	

[CT006] Reservar Vaga de um estacionamento	
Descrição: O responsável pelo teste deve tentar reservar uma vaga em algum dos estacionamentos retornados na pesquisa.	
Requisito(s) Associado(s): RF08	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar com o GPS ativado e conectado à rede Wifi. 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve ter acesso a página de vagas do estacionamento e ao escolher uma das vagas com um click longo a opção de reservar àquela vaga aparecerá, ao clicar em reservar a cor da vaga deve mudar para Amarela e uma notificação deve ser enviada ao dono do estacionamento, o qual poderá cancelar a reserva se desejar.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • A reserva deve ser cadastrada no banco e ficar disponível para o dono do estacionamento. 	
Procedimento Associado: [PT006]	

2.2 Usuário Gestor do Estacionamento

[CT007] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo Gestor de Estacionamento.	
Requisito(s) Associado(s): RF01	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone utilizado. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve ser cadastrado no sistema. • As credenciais devem ser cadastradas no <i>Firebase</i>. • O novo nó com o registro do usuário deve ser salvo no ramo de Usuários no <i>Realtime Database</i>.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • Uma mensagem de confirmação deve ser exibida caso as informações inseridas estejam corretas. 	
Procedimento Associado: [PT007]	

[CT008] Autenticar-se no Sistema

Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu email e senha cadastrados

Requisito(s) Associado(s):

Pré Condições:

- Possuir o aplicativo instalado no smartphone.
- Possuir cadastro no sistema.
- Estar conectado à rede Wifi

Pós Condições:

- O usuário deve acessar a página inicial do aplicativo e ter acesso a todas as opções que o app oferece de acordo com o tipo de usuário do cadastro.

Critérios de Sucesso:

- Pós condições satisfeitas
- Um *Toast* informando “Operação realizada com sucesso” deve aparecer na tela.

Procedimento Associado: [PT008]

[CT009] Atualizar informações do Cadastro

Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.

Requisito(s) Associado(s): RF01

Pré Condições:

- Possuir o aplicativo instalado no smartphone.
- Possuir cadastro no sistema.
- Estar conectado à rede Wifi

Pós Condições:

- O usuário deve ter acesso a página de edição com suas informações previamente preenchidas de acordo com o que foi informado no cadastro inicial.

Critérios de Sucesso:

- Pós condições satisfeitas
- Um *Toast* informando “Operação realizada com sucesso” deve aparecer na tela.

Procedimento Associado: [PT009]

[CT010] Gerenciar Estacionamentos	
Descrição: O responsável pelo teste deve tentar cadastrar um estacionamento no sistema.	
Requisito(s) Associado(s): RF02	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve ter acesso a uma página principal com as informações do estacionamento definido como atual e quando necessário editar as informações do estacionamento assim como adicionar novos estacionamentos que podem ser definidos como atual quando necessário. Também é possível, editar, excluir e cadastrar vagas para o estacionamento em questão.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • A cada novo estacionamento cadastrado, este aparece na lista de estacionamentos cadastrados. • Ao clicar em um carro da lista as informações deste deve aparecer na página principal como o carro atual escolhido. 	
Procedimento Associado: [PT010]	

[CT011] Gerenciar Vagas do Estacionamento	
Descrição: O responsável pelo teste deve cadastrar vagas assim como atualizar seu status de acordo com a utilização.	
Requisito(s) Associado(s): RF05	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • Ao clicar em adicionar vagas na barra de ações do estacionamento, acionada ao clique longo em determinado estacionamento, o usuário deve ter acesso a uma tela com as vagas cadastradas, podendo adicionar ou excluir uma vaga.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas • Na tela principal do estacionamento ao clicar em gerenciar vagas o usuário deve ter acesso a tela de vagas exibindo o status de cada. 	
Procedimento Associado: [PT011]	

[CT012] Gerenciar Reservas de vaga de um estacionamento	
Descrição: O responsável pelo teste deve aceitar as solicitações de reservas de vagas recebidas.	
Requisitos Associados: RF08	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar com o GPS ativado. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve poder cancelar a reserva quando desejar, isso é possível através de um clique longo na reserva.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas 	
Procedimento Associado: [PT012]	

[CT013] Recuperação de senha	
Descrição: O responsável pelo teste deve tentar recuperar a senha.	
Requisitos Associados: RF01	
Pré Condições: <ul style="list-style-type: none"> • Possuir o aplicativo instalado no smartphone. • Possuir cadastro no sistema. • Estar conectado à rede Wifi 	Pós Condições: <ul style="list-style-type: none"> • O usuário deve poder cadastrar uma nova senha e acessar o sistema com a nova senha cadastrada.
Critérios de Sucesso: <ul style="list-style-type: none"> • Pós condições satisfeitas 	
Procedimento Associado: [PT013]	

3. Procedimentos de Teste

Um procedimento de teste é composto por passos que descrevem como um determinado teste será configurado e como deve ser executado. Os procedimentos de testes estão descritos nos quadros abaixo.

3.1 Usuário Motorista

[PT001] Cadastro do Usuário

Descrição: procedimentos e requisitos necessários para realizar o cadastro de um usuário do tipo motorista no sistema.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Selecione a opção “não tem conta? Cadastre-se” na tela inicial do aplicativo;
- Preencha os campos de nome, e-mail e senha na tela de cadastro com valores válidos;
- Escolha o tipo Motorista na seção Tipo de Usuário
- Clique no botão “Cadastrar”.

[PT002] Autenticar-se no Sistema

Descrição: procedimentos e requisitos necessários para se autenticar no sistema e acessar as funcionalidades do aplicativo.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Preencha o campo e-mail com o e-mail do usuário cadastrado;
- Insira a senha no campo senha;
- Pressione o botão “Logar”.

[PT003] Atualizar Informações do Cadastro

Descrição: procedimentos e requisitos necessários para atualizar as informações do cadastro do usuário.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral;
- Escolha a opção Alterar perfil, uma tela com as informações do perfil do usuário aparecerá;
- Escolha o ícone editar presente nesta tela ao lado do nome do usuário. Uma nova tela será exibida com os campos relacionados ao cadastro do usuário do tipo motorista e com as informações fornecidas anteriormente;
- Preencha os campos em branco ou edite os campos desejados com nome, email, telefone, endereço e número da Carteira Nacional de Habilitação.
- Pressione o botão “Salvar”.

[PT004] Gerenciar veículos

Descrição: Procedimentos e requisitos necessários para a realização do cadastro, edição e exclusão de um veículo no sistema.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral;
- Escolha a opção Gerenciar Veículo, uma tela com as informações do veículo que está sendo utilizado no momento aparecerá;
- Escolha o ícone atualizar (Floating Button) presente nesta tela ao lado do número da placa do veículo, uma nova tela será exibida com a lista de veículos cadastrados no sistema ;
- Para adicionar um novo veículo
 - Clique no botão flutuante com o símbolo de + (mais) presente na parte inferior da tela;
 - Preencha os campos do modelo, fabricante, ano, cor e placa que aparecerá em uma janela de diálogo;
 - Clique no botão “Done” no teclado virtual do smartphone depois de preencher o último campo.
- Para editar um veículo existente
 - Clique e segure no veículo que deseja editar, um menu de ações aparecerá;
 - Selecione o ícone de edição (com a figura de um lápis);
 - Edite as informações desejadas na janela de diálogo que aparecerá;
 - Clique no botão “Done” no teclado virtual do smartphone para salvar as alterações (O botão Done aparecerá quando o usuário passar por todos os campos e estiver no campo da placa do veículo).
- Para excluir um carro existente
 - Clique e segure no carro que deseja excluir, um menu de ações aparecerá;
 - Selecione o ícone de excluir (com a figura de uma lixeira);
 - Pronto! O estacionamento foi excluído da lista.

[PT005] Procurar Vagas para Estacionar

Descrição: Procedimento e requisitos necessários para busca de vagas e seleção de um estacionamento de interesse com vagas disponíveis.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- A tela de busca de vagas é a tela principal do aplicativo, logo a mesma já deverá aparecer assim que o usuário entrar no sistema, porém se o usuário estiver na tela de outra opção do menu, abra o menu lateral e escolha a opção buscar Vagas, um mapa com a localização atual do usuário aparecerá;
- Digite o endereço de destino no campo de pesquisa e clique no ícone de pesquisa (com a figura de uma lupa), uma janela de diálogo com as opções de endereço com descrição semelhante ao endereço digitado ou até apenas com a opção do endereço digitado aparecerá;
- Selecione o endereço desejado;
- O aplicativo exibirá todos os estacionamentos que existem em um raio de 1 km;
- Escolha o estacionamento desejado;
- Um diálogo aparecerá com as informações do estacionamento e duas opções: rota e vagas.
- Ao escolher a opção “rota” o aplicativo exibirá a rota que o motorista terá que realizar do seu local atual até o endereço de destino.
- Ao escolher Vagas o sistema redireciona o usuário para a página de gerenciamento de vagas onde é possível reservar uma vaga.

[PT006] Reservar vaga de estacionamento

Descrição: Procedimentos e requisitos necessário para realizar a reserva de uma vaga.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Ainda na tela da opção Buscar Vagas, depois de escolhido o endereço de destino, clique no estacionamento desejado, uma tela de informações do estacionamento aparecerá.
- Escolha a opção vagas;
- Uma nova tela com as vagas preenchidas com a cor relativa a seu status aparecerá.
- Escolha uma vaga que esteja na cor verde, clique nesta e segure;
- Um menu de ações aparecerá, escolha a opção reservar vaga (ícone amarelo);
- Volte para o mapa e escolha a opção rota para que o aplicativo exiba a rota que o motorista terá que realizar do seu local atual até o endereço de destino.

3.2 Usuário Gestor do Estacionamento

[PT007] Cadastrar Usuário

Descrição: Procedimentos e requisitos para o cadastro de um usuário do tipo gestor de estacionamento no sistema.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Selecione a opção “não tem conta? Cadastre-se” na tela inicial do aplicativo;
- Preencha os campos de nome, email e senha na tela de cadastro com valores válidos;
- Escolha o tipo Gestor de Estacionamento na seção Tipo de Usuário
- Clique no botão “Cadastrar”.

[PT008] Autenticar-se no sistema

Descrição: Procedimentos e requisitos necessário para autenticar-se no sistema e acessar as funcionalidades do aplicativo.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Preencha o campo e-mail com o e-mail do usuário cadastrado;
- Insira a senha no campo senha;
- Pressione o botão “Logar”.

[PT009] Atualizar informações do Cadastro

Descrição: Procedimentos e requisitos para atualização do cadastro do usuário do tipo gestor de estacionamento logado no sistema.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral;
- Escolha a opção Alterar perfil, uma tela com as informações do perfil do usuário aparecerá;
- Escolha o ícone editar presente nesta tela ao lado do nome do usuário (Floating Button com o ícone de um lápis). Uma nova tela será exibida com os campos relacionados ao cadastro do usuário do tipo gestor de estacionamento e com as informações fornecidas anteriormente preenchendo o campo o qual corresponde;
- Preencha os campos em branco ou edite os campos desejados com nome, email, telefone e número da CNH;
- Pressione o botão “Salvar”.

[PT010] Gerenciar Estacionamentos

Descrição: Procedimentos e requisitos necessários para a realização do cadastro, edição e exclusão de um estacionamento no sistema.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral;
- Escolha a opção Gerenciar Estacionamento, uma tela com as informações do estacionamento que está sendo utilizado no momento aparecerá;
- Escolha o botão flutuante com o ícone atualizar presente nesta tela ao lado do nome do estacionamento, uma nova tela será exibida com a lista de estacionamentos cadastrados no sistema;
- Para adicionar um novo estacionamento
 - Clique no botão flutuante com o símbolo de + (mais) presente na parte inferior da tela;
 - Preencha os campos do cnpj, nome do estacionamento, endereço e telefone que aparecerá em uma janela de diálogo;
 - Clique no botão "Done" no teclado virtual do smartphone depois de preencher o último campo (O botão Done aparecerá quando o usuário preencher todos os campos e estiver no campo endereço). O sistema fará uma busca dos endereços com descrição semelhante ao endereço digitado.
 - Um diálogo com os endereços encontrados aparecerá. escolha o endereço desejado.
- Para editar um estacionamento existente
 - Clique e segure no estacionamento que deseja editar, um menu de ações aparecerá;
 - Selecione o ícone de edição (com a figura de um lápis);
 - Edite as informações desejadas na janela de diálogo que aparecerá;
 - Salve as alterações clicando no botão "Done" no teclado virtual do smartphone (O botão Done aparecerá quando o usuário preencher todos os campos e estiver no campo endereço). O sistema fará uma busca dos endereços com descrição semelhante ao endereço digitado.
- Para excluir um estacionamento existente
 - Clique e segure no estacionamento que deseja excluir, um menu de ações aparecerá;
 - Selecione o ícone de excluir (com a figura de uma lixeira);
 - Pronto, o estacionamento foi excluído da lista.

[PT011] Gerenciar Vagas

Descrição: Procedimentos e requisitos necessários para o adicionar, mudar o status e excluir uma vaga em um determinado estacionamento.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral e escolha a opção Gerenciar Estacionamento;
- Escolha o ícone ao lado do nome do estacionamento, uma lista com todos os estacionamentos aparecerá;
- Escolha o estacionamento o qual deseja gerenciar as vagas, neste clique e segure, um menu de ações aparecerá;
- Neste menu escolha o ícone de adicionar vagas (símbolo de mais envolto em um círculo), uma nova tela será exibida com as vagas cadastradas.
- Para adicionar uma nova vaga.
 - Clique no botão flutuante com o símbolo de + (mais) presente na parte inferior da tela;
 - Uma nova vaga aparecerá na lista de vagas;
- Para excluir uma vaga
 - Escolha a vaga, clique e segure
 - Pronto! a vaga será excluída da lista;
- Para mudar o status de uma vaga
 - Na tela principal de Gerenciar estacionamento escolha a opção Gerenciar Vagas. O usuário será redirecionado para a tela de gerenciamento de vagas.
 - Escolha a vaga, clique e segure, um menu de opções aparecerá;
 - Escolha a opção de status desejada (ícones coloridos representado seu respectivo status, vermelho - em uso, amarelo - reservado, verde - livre);
 - Se o usuário tentar modificar o status de uma vaga reservada uma mensagem de alerta aparecerá impossibilitando que o usuário execute a ação.

[PT012] Gerenciar Reservas de vaga de um estacionamento

Descrição: Procedimentos e requisitos para gerenciar solicitações de reserva de vagas.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Abra o menu lateral e escolha a opção Gerenciar Reservas, uma lista com todas as solicitações de reserva aparecerá;
- Selecione uma das solicitações, clique e segure.
- Uma mensagem perguntando se deseja cancelar a reserva aparecerá.
- Ao clicar sim a reserva é cancelada e a vaga volta a ficar livre.
- Ao clicar não, nada acontece e a janela desaparece.

[PT013] Recuperar senha

Descrição: Procedimentos e requisitos recuperar a senha de acesso.

Configuração inicial do ambiente:

- Configuração do(s) smartphone (s): Android 4.0 (Ice Cream Sandwich) API Level 15. Qualcomm Snapdragon 200 MSM8212 Cortex-A7. 1.2 GHz Quad Core
- Sensores Necessários: GPS;

Passos:

- Na tela de login escolha a opção “Esqueci a senha”.
- Um diálogo aparecerá, neste digite o e-mail cadastrado no sistema.
- Uma mensagem será enviada para o e-mail informado com um link para a recuperação da senha.

4. Sumário de Avaliação de Testes – Testador 1

Esta seção descreve o teste realizado pelo Testador 1. O Testador 1 é estudante do último período de Ciência da Computação na Universidade Federal de Sergipe e utilizou um LG K2 Titânio, com Processador Octacore 1.5 Ghz - Mediatek MT6750 com memória interna de 32Gb, 2 Gb de RAM e Android 7.0 durante a realização dos testes.

4.1 Usuário Motorista**[SAT001] Cadastro do Usuário**

Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo motorista.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para a realização do teste.

Critérios de Sucesso:

- Exibida mensagem de cadastro efetuado com sucesso.

Resultado Obtido: Usuário cadastrado.

[SAT002] Autenticar-se no Sistema	
Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu email e senha cadastrados	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Exibida mensagem de operação realizada com sucesso. 	
Resultados Obtidos: Usuário conseguiu acessar o sistema.	

[SAT003] Atualizar informações do Cadastro	
Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Exibida mensagem de operação realizada com sucesso; Novas informações foram exibidas no perfil do usuário. 	
Resultados Obtidos: Dados do usuário foram atualizados.	
Observação: O nome real da opção é “Alterar Perfil”. Corrigir [PT003].	

[SAT004] Cadastrar veículos	
Descrição: O responsável pelo teste deve tentar cadastrar um veículo no sistema.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Foram exibidas mensagens de sucesso para todas as operações; Novos veículos apareceram na lista de veículos cadastrados; Veículos excluídos foram removidos da lista de veículos cadastrados; Novas informações foram corretamente atualizadas sobre os veículos após edição; Ao clicar em um carro da lista as informações deste deve apareceram na página principal como o carro atual escolhido. 	
Resultados Obtidos: Veículos cadastrados, editados e removidos com sucesso. Observação: Após abrir a página do carro atual escolhido, ao retornar para a tela de Gerenciar Veículos, a lista parece vazia. Apenas clicando no botão flutuante com o símbolo de + (mais) para adicionar outro veículo a lista é recarregada. Verificar [PT004]. No último passo para excluir um carro existente, diz que “o estacionamento foi excluído da lista”.	

[SAT005] Procurar vagas para estacionar	
Descrição: O responsável pelo teste deve informar um local de destino e o sistema irá retornar todos os estacionamentos próximos com vagas em um raio de 1 Km.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições não foram satisfatórias para realização do teste. Pois também é preciso que haja Estacionamentos e Vagas cadastrados no sistema. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Usuário conseguiu selecionar estacionamento próximo ao endereço de destino, assim como reservar vaga e traçar rota. 	
Resultados Obtidos: Estacionamento e vagas encontradas. Observação: Usuário não conseguiu sair do modo de rota para realizar outra busca por estacionamentos em outro endereço.	

[SAT006] Reservar Vaga de um estacionamento	
Descrição: O responsável pelo teste deve tentar reservar uma vaga em algum dos estacionamentos retornados na pesquisa.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições não foram satisfatórias para realização do teste. Pois também é preciso que haja Estacionamentos e Vagas cadastrados no sistema. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Status da vaga alterado para reservada. 	
Resultados Obtidos: Vaga reservada com sucesso.	

4.2 Usuário Gestor do Estacionamento

[SAT007] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo Gestor de Estacionamento.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Exibida mensagem de sucesso. 	
Resultados Obtidos: Usuário cadastrado. Observação: Em [PT007], é dito para escolher tipo Motorista na seção Tipo de Usuário, o que não condiz com a descrição do procedimento. Neste teste, foi escolhida a opção Gestor de Estacionamento.	

[SAT008] Autenticar-se no Sistema

Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu e-mail e senha cadastrados

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Exibida mensagem de sucesso;
- Carregada página inicial do sistema.

Resultados Obtidos: Autenticação efetuada no sistema.

[SAT009] Atualizar informações do Cadastro

Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Exibida mensagem de operação realizada com sucesso;
- Novas informações foram exibidas no perfil do usuário.

Resultados Obtidos: Dados do usuário foram atualizados.

Observação: O nome real da opção é "Alterar Perfil". Corrigir [PT009].

[SAT010] Gerenciar Estacionamentos

Descrição: O responsável pelo teste deve tentar cadastrar um estacionamento no sistema.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições não foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Estacionamentos cadastrados e mensagens de sucesso exibidas;
- Ao clicar em um estacionamento, lista de informações é exibida.

Resultados Obtidos: Estacionamentos cadastrados e editados com sucesso. Entretanto, o processo de exclusão falha: estacionamentos excluídos voltam para a lista. Não há mensagem de sucesso, o estacionamento desaparece da lista, mas retorna após ser realizada outra operação no sistema.

Observação: Em [CT010], um dos Critérios de Sucesso é clicar em um carro da lista as informações para aparecer na página principal como o carro atual escolhido, o que não condiz com o teste. Neste teste, uma operação análoga foi realizada com Estacionamento.

[SAT011] Gerenciar Vagas do Estacionamento

Descrição: O responsável pelo teste deve cadastrar vagas assim como atualizar seu status de acordo com a utilização.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Acesso à tela com as vagas cadastradas, podendo adicionar ou excluir vagas;
- Acesso à tela de Gerenciamento de Vagas com o status de cada.

Resultados Obtidos: Vagas adicionadas, excluídas e com status alterados com sucesso.

Observação: O passo 2 do [PT011] não condiz com o procedimento.

[SAT012] Gerenciar Reservas de vaga de um estacionamento

Descrição: O responsável pelo teste deve aceitar as solicitações de reservas de vagas recebidas.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Com um clique longo sobre a reserva, a reserva foi cancelada e a vaga liberada.

Resultados Obtidos: Reserva cancelada.

[SAT013] Recuperação de senha

Descrição: O responsável pelo teste deve tentar recuperar a senha.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pré-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Recebido e-mail para redefinir senha;
- Senha redefinida e autenticada com sucesso.

Resultados Obtidos: Senha recuperada.

5. Sumário de Avaliação de Testes – Testador 2

Esta seção descreve o teste realizado pelo Testador 2. O Testador 2 é estudante do curso de Ciência da Computação na Universidade Federal de Sergipe e utilizou um Samsung Galaxy J5 Metal com Android 6.0.1, Processador 1.2 GHz Quad Core e 2GB de RAM durante a realização dos testes.

5.1 Usuário Motorista

[SAT001] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo motorista.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultado Obtido: Cadastrado realizado. Observação: O campo “nome” está aceitando apenas números.	

[SAT002] Autenticar-se no Sistema	
Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu e-mail e senha cadastrados	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Acesso ao sistema permitido.	

[SAT003] Atualizar informações do Cadastro	
Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Dados atualizados. Observação: O procedimento [PT003] aponta a opção “Atualizar perfil”, mas no aplicativo o nome é “Alterar perfil”.	

[SAT004] Cadastrar veículos	
Descrição: O responsável pelo teste deve tentar cadastrar um veículo no sistema.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados 	
Resultados Obtidos: Sucesso ao cadastrar, editar e remover veículo.	

[SAT005] Procurar vagas para estacionar	
Descrição: O responsável pelo teste deve informar um local de destino e o sistema irá retornar todos os estacionamentos próximos com vagas em um raio de 1 Km.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. Pois também é preciso que haja Estacionamentos e Vagas cadastrados no sistema. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Foram encontrados estacionamento, rotas e vagas. Observação: Depois de traçar uma rota, não foi possível pesquisar outra localização.	

[SAT006] Reservar Vaga de um estacionamento	
Descrição: O responsável pelo teste deve tentar reservar uma vaga em algum dos estacionamentos retornados na pesquisa.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições não foram satisfatórias para realização do teste. Necessário a existência de Estacionamentos e Vagas cadastrados. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Vaga reservada com sucesso.	

5.2 Usuário Gestor do Estacionamento

[SAT007] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo Gestor de Estacionamento.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultado Obtido: Cadastrado realizado. Observação: O campo “nome” está aceitando apenas números.	

[SAT008] Autenticar-se no Sistema	
Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu e-mail e senha cadastrados	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Acesso ao sistema permitido.	

[SAT009] Atualizar informações do Cadastro	
Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Dados atualizados. Observação: O procedimento [PT009] aponta a opção “Atualizar perfil”, mas no aplicativo o nome é “Alterar perfil”.	

[SAT010] Gerenciar Estacionamentos	
Descrição: O responsável pelo teste deve tentar cadastrar um estacionamento no sistema.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições não foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso parcialmente alcançados. 	
Resultados Obtidos: Cadastro e edição realizados com sucesso. Ao realizar a exclusão, estacionamento desaparece mas volta a aparecer na lista. Observação: Na lista de estacionamentos, o nome do estacionamento aparece por cima do endereço dificultando a visualização do endereço	

[SAT011] Gerenciar Vagas do Estacionamento	
Descrição: O responsável pelo teste deve cadastrar vagas assim como atualizar seu status de acordo com a utilização.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Sucesso em adicionar, excluir e alterar status das vagas. Observação: O passo 2 do procedimento [PT011] não foi realizado. A opção citada não existe.	

[SAT012] Gerenciar Reservas de vaga de um estacionamento	
Descrição: O responsável pelo teste deve aceitar as solicitações de reservas de vagas recebidas.	
Requisitos Associados: N/A	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Reserva cancelada.	

[SAT013] Recuperação de senha	
Descrição: O responsável pelo teste deve tentar recuperar a senha.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultados Obtidos: Acesso ao sistema com a nova senha.	

6. Sumário de Avaliação de Testes – Testador 3

Esta seção descreve o teste realizado pelo Testador 3. O Testador 3 é estudante do curso de Ciência da Computação na Universidade Federal de Sergipe e utilizou um Motorola Moto E 2ª Geração com Android 5.1 e Processador de 1.2 GHz quad-core Qualcomm Snapdragon 200 (MSM8212) com Memória RAM de 1GB durante a realização dos testes.

6.1 Usuário Motorista

[SAT001] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo motorista.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para a realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados. 	
Resultado Obtido: Usuário Cadastrado Observações: É possível criar um usuário com o nome contendo apenas números	

[SAT002] Autenticar-se no Sistema

Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu e-mail e senha cadastrados

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para a realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados.

Resultados Obtidos: Usuário conseguiu acesso ao sistema

[SAT003] Atualizar informações do Cadastro

Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados.

Resultados Obtidos: Dados alterados com sucesso.

Observações: Telefone e CNH não válidos foram aceitos, contendo apenas 3 números por exemplo.

[SAT004] Cadastrar veículos	
Descrição: O responsável pelo teste deve tentar cadastrar um veículo no sistema.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados 	
Resultados Obtidos: Veículos cadastrado com sucesso	

[SAT005] Procurar vagas para estacionar	
Descrição: O responsável pelo teste deve informar um local de destino e o sistema irá retornar todos os estacionamentos próximos com vagas em um raio de 1 Km.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados 	
Resultados Obtidos: Estacionamento e vagas foram encontrados	

[SAT006] Reservar Vaga de um estacionamento	
Descrição: O responsável pelo teste deve tentar reservar uma vaga em algum dos estacionamentos retornados na pesquisa.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições não foram satisfatórias para realização do teste, são necessários estacionamentos e vagas cadastrados no sistema. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados 	
Resultados Obtidos: Vaga reservada com sucesso	

6.2 Usuário Gestor do Estacionamento

[SAT007] Cadastro do Usuário	
Descrição: O responsável pelo teste deve tentar cadastrar um usuário no sistema do tipo Gestor de Estacionamento.	
Pré Condições: <ul style="list-style-type: none">As pré-condições foram satisfatórias para realização do teste.	Pós Condições: <ul style="list-style-type: none">As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none">Critérios de sucesso alcançados	
Resultados Obtidos: É possível criar um usuário com o nome contendo apenas números.	

[SAT008] Autenticar-se no Sistema	
Descrição: O responsável pelo teste deve tentar se autenticar no sistema digitando seu e-mail e senha cadastrados	
Pré Condições: <ul style="list-style-type: none">As pré-condições foram satisfatórias para realização do teste.	Pós Condições: <ul style="list-style-type: none">As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none">Critérios de sucesso alcançados	
Resultados Obtidos: Usuário conseguiu acesso ao sistema	

[SAT009] Atualizar informações do Cadastro

Descrição: O responsável pelo teste deve tentar modificar suas informações de cadastro assim como preencher campos que ainda não foram informados.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados

Resultados Obtidos: Dados alterados com sucesso.

Observações: Telefone e CNH não válidos foram aceitos, contendo apenas 3 números por exemplo.

[SAT010] Gerenciar Estacionamentos

Descrição: O responsável pelo teste deve tentar cadastrar um estacionamento no sistema.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados

Resultados Obtidos: Estacionamento cadastrado com sucesso.

Observações: Falha ao excluir estacionamentos, ele volta para a lista

[SAT011] Gerenciar Vagas do Estacionamento

Descrição: O responsável pelo teste deve cadastrar vagas assim como atualizar seu status de acordo com a utilização.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados

Resultados Obtidos: Vagas adicionadas e excluídas com sucesso, assim como seus status também são alterados, única observação para vagas marcadas como “em uso”, após sair da tela elas voltam para o status “Livre”

[SAT012] Gerenciar Reservas de vaga de um estacionamento

Descrição: O responsável pelo teste deve aceitar as solicitações de reservas de vagas recebidas.

Pré Condições:

- As pré-condições foram satisfatórias para realização do teste.

Pós Condições:

- As pós-condições foram satisfatórias para realização do teste.

Critérios de Sucesso:

- Critérios de sucesso alcançados

Resultados Obtidos: Reservas foram canceladas com sucesso

[SAT013] Recuperação de senha	
Descrição: O responsável pelo teste deve tentar recuperar a senha.	
Pré Condições: <ul style="list-style-type: none"> As pré-condições foram satisfatórias para realização do teste. 	Pós Condições: <ul style="list-style-type: none"> As pós-condições foram satisfatórias para realização do teste.
Critérios de Sucesso: <ul style="list-style-type: none"> Critérios de sucesso alcançados 	
Resultados Obtidos: Senha alterada com sucesso	

APÊNDICE G – Manual do Usuário do Smart Parking

Abaixo segue o manual para os usuários do Smart Parking, o qual descreve o funcionamento de cada opção presente no aplicativo, guiando o usuário no que deve ser realizado em cada tela para alcançar o objetivo desejado.



Smart Parking – Manual do Usuário

Sumário

1. Acesso ao Sistema.....	3
2. Cadastro no Sistema.....	3
3. Recuperação de senha.....	4
4. Buscar Vagas (Tela Inicial do motorista).....	5
5. Gerenciamento de Veículos	10
6. Atualizar Perfil.....	12
7. Tela de Gerenciamento do Estacionamento.....	14
8. Gerenciamento de reservas.....	16

O Smart Parking é um aplicativo voltado para o gerenciamento da oferta e procura de estacionamentos em geral. O presente documento descreve o funcionamento de cada opção presente no aplicativo, guiando o usuário no que deve ser realizado em cada tela para alcançar o objetivo desejado.

1. Acesso ao Sistema

Ao abrir o aplicativo a primeira tela apresentada é a tela de *Login*, nesta é possível se autenticar no sistema, caso já possua uma conta cadastrada ou realizar o cadastro. Também é possível recuperar a senha de um usuário cadastrado clicando na opção “Esqueci a senha”.



Figura 1 – Página inicial do aplicativo

2. Cadastro no Sistema

Ao escolher a opção “Não tem conta? Cadastre-se” o usuário é direcionado para a tela de cadastro (Figura 2), na qual o mesmo pode inserir as informações solicitadas e criar uma conta de acesso ao sistema do aplicativo.

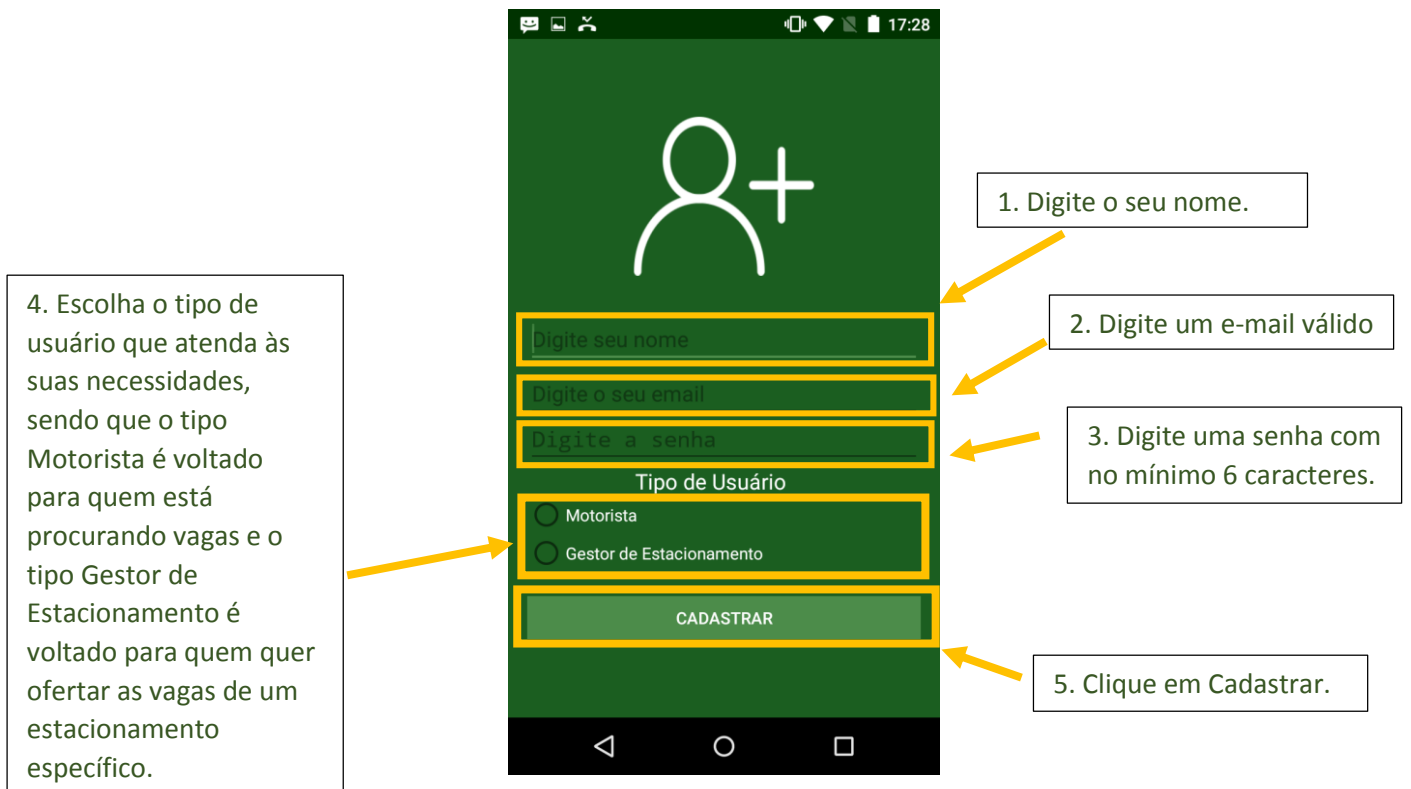


Figura 2 – Tela de Cadastro o Usuário

Ao clicar em cadastrar, se todas as informações forem inseridas corretamente, uma mensagem informando que a operação foi realizada com sucesso aparecerá, caso contrário aparecerá uma mensagem informando o erro que ocorreu.

3. Recuperação de senha

Nos casos em que o usuário esquece a senha, a opção “Esqueci a senha” deve ser escolhida na tela de login e a caixa de diálogo exibido na Figura 3 irá aparecer.

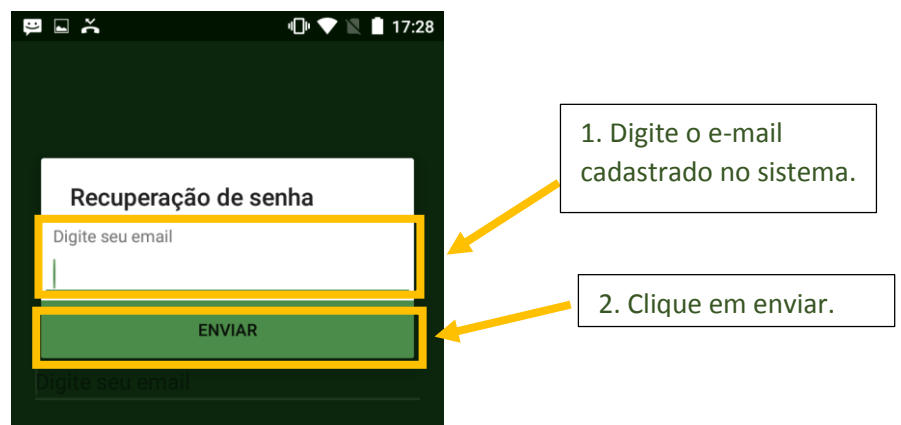


Figura 3 – Caixa de diálogo para recuperação de senha

Uma mensagem será enviada para seu e-mail com um link, ao clicar neste link você será redirecionado para uma página, onde cadastrará sua nova senha.

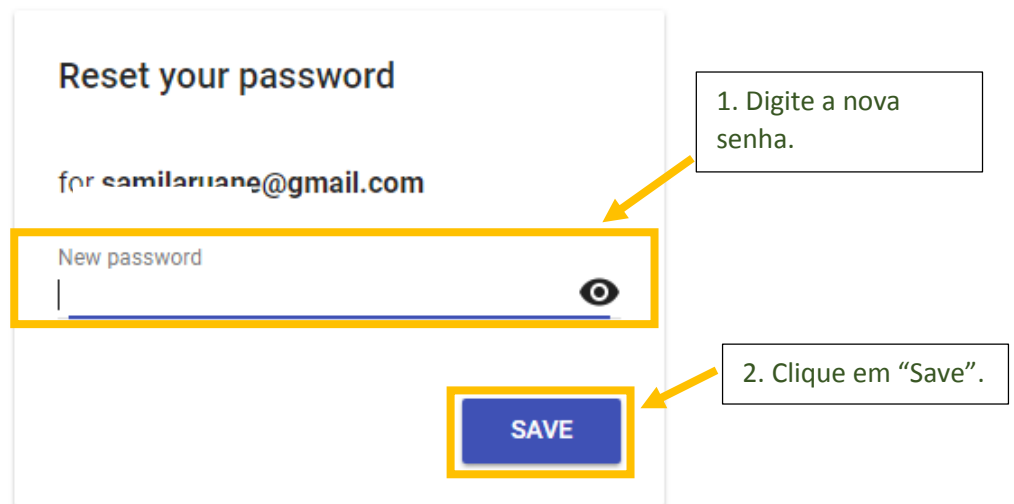


Figura 4 – Recuperação de senha

Depois de realizado o procedimento acima, o usuário pode acessar o sistema com a nova senha cadastrada.

4. Buscar Vagas (Tela Inicial do motorista)

Ao se autenticar com um usuário do tipo motorista, a tela inicial que será exibida conterá um mapa com a localização atual do usuário. A partir deste mapa é possível fazer a busca por estacionamentos próximos ao destino desejado.

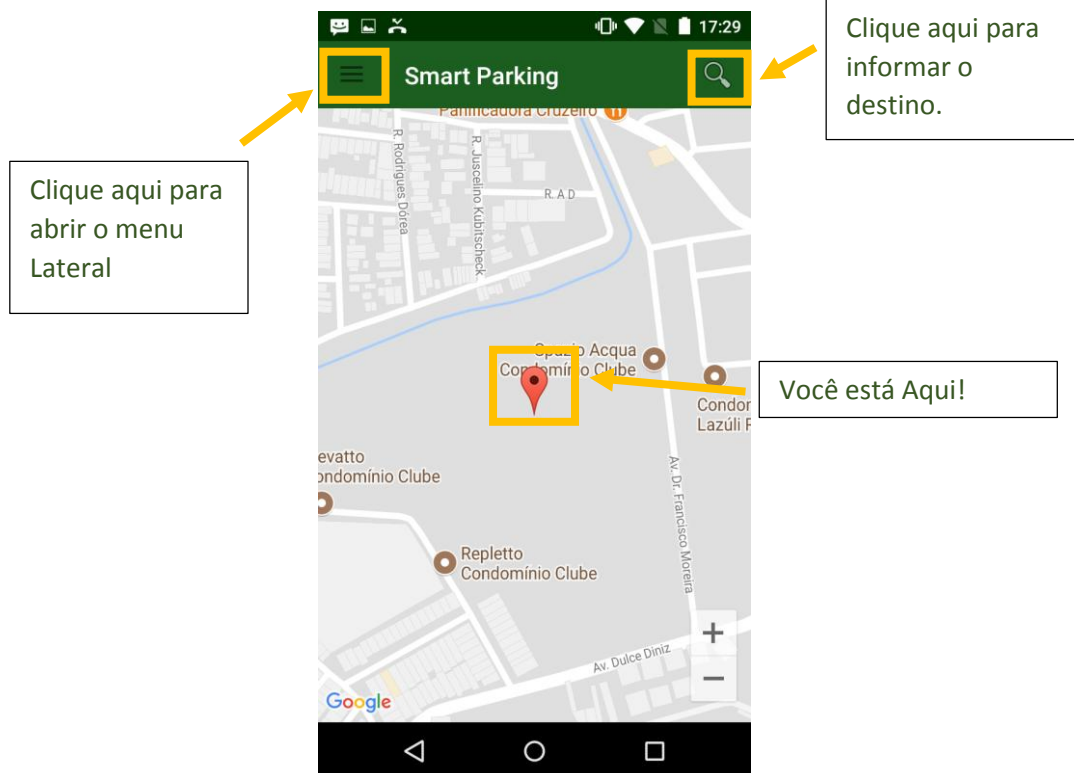
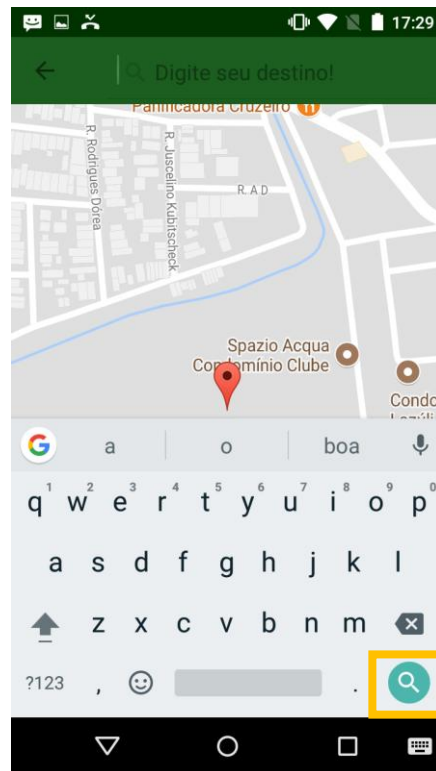


Figura 5 – Página inicial do motorista

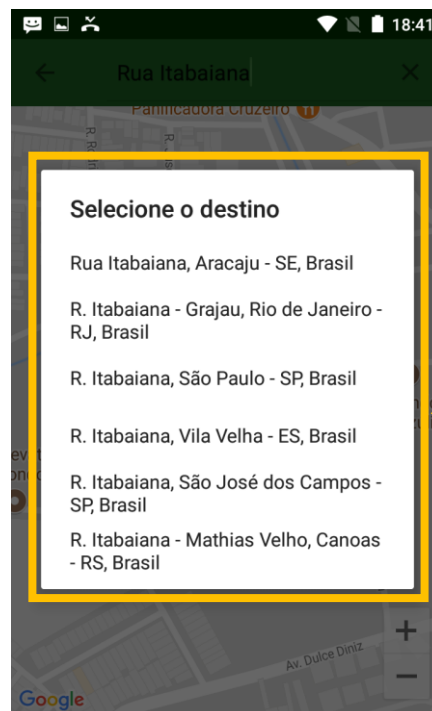
Ao clicar na Lupa, o teclado virtual do smartphone irá aparecer, neste digite o endereço de destino.



Depois de digitar o endereço de destino clique na lupa para iniciar a pesquisa.

Figura 6 – Teclado Virtual

Ao clicar na lupa do teclado todos os endereços semelhantes ao digitado aparecerá em uma lista. Escolha aquele que se refere ao endereço desejado.



Escolha o destino desejado.

Figura 7 – Caixa de diálogo com os endereços semelhantes ao digitado

Ao escolher o endereço desejado todos os estacionamentos existentes a uma distância de 1 km do local informado aparecerão.

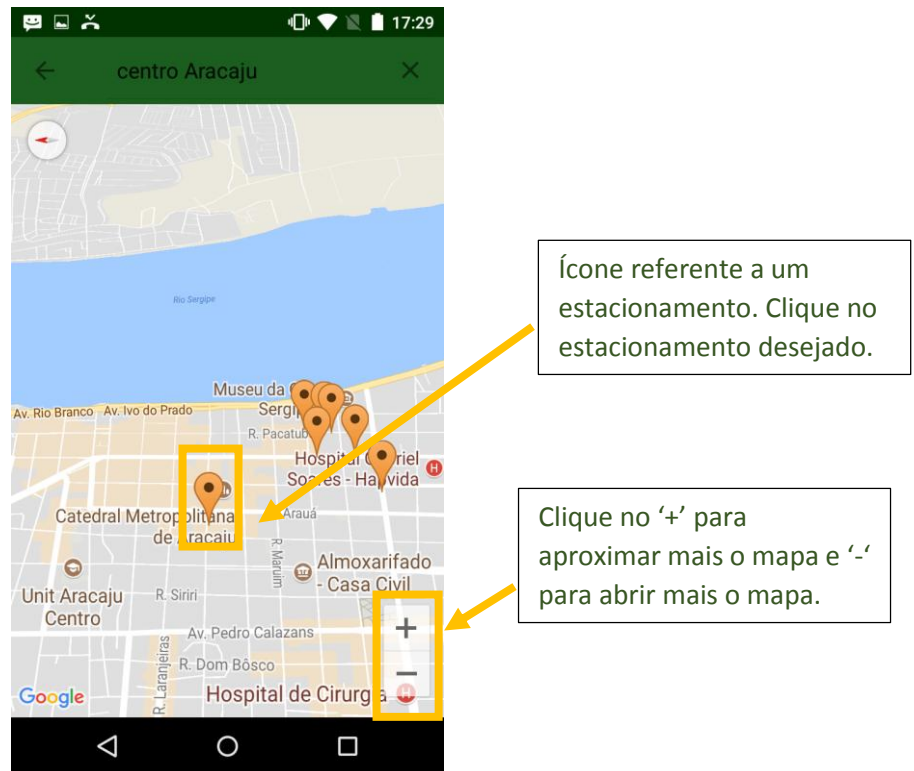


Figura 8 – Estacionamentos localizados próximo ao destino

Ao clicar em algum dos estacionamentos uma caixa de diálogo aparecerá contendo as informações do estacionamento escolhido.

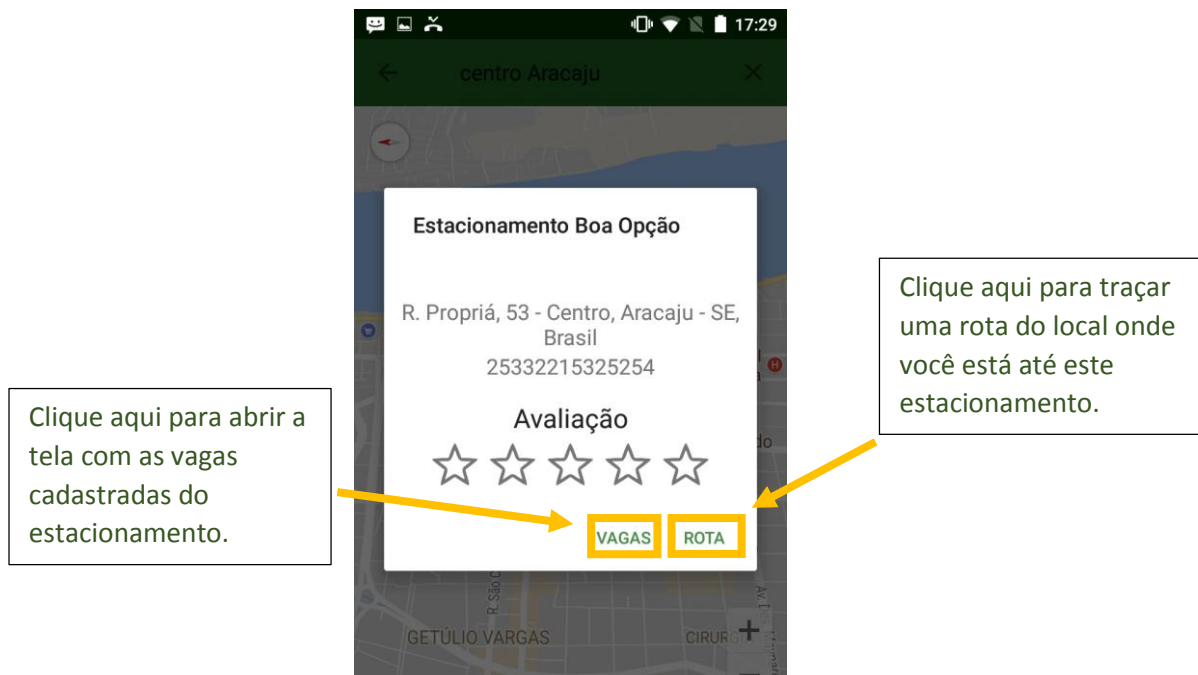


Figura 9 – Caixa de diálogo com as informações do estacionamento

Ao clicar em “Vagas” o usuário será redirecionado a tela de vagas do estacionamento escolhido.

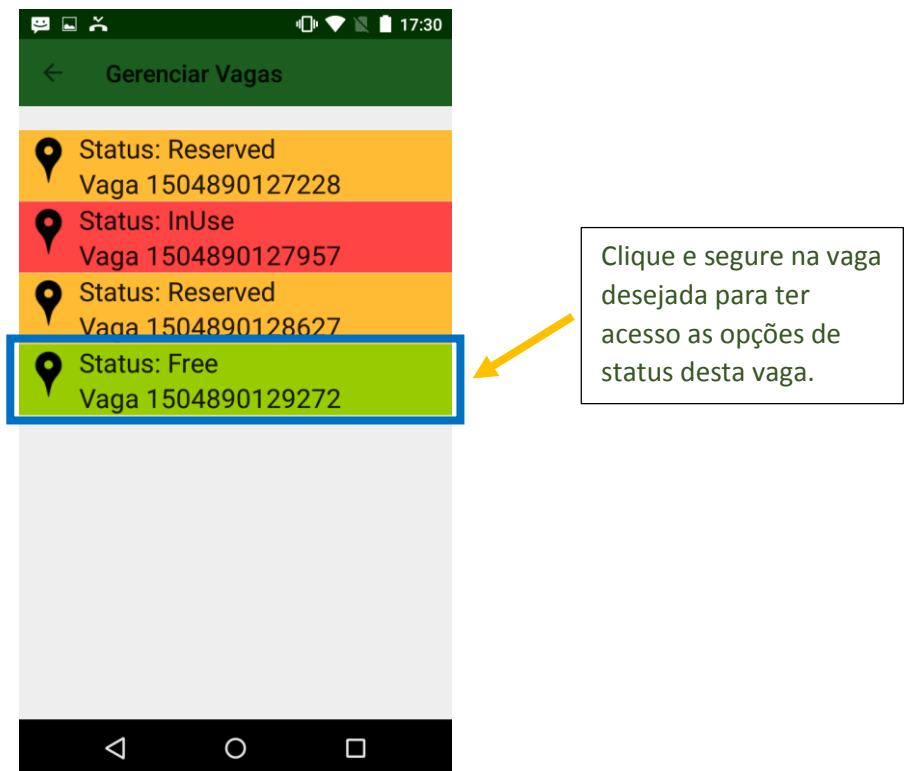


Figura 10 – Tela de Vagas do estacionamento

A cor de cada vaga remete ao status atual da mesma. As vagas em vermelho são as ocupadas, as vagas em amarelo são as reservadas e as vagas em verde são as livres. Ao clicar e segurar em uma vaga um menu com opções de status aparecerá.

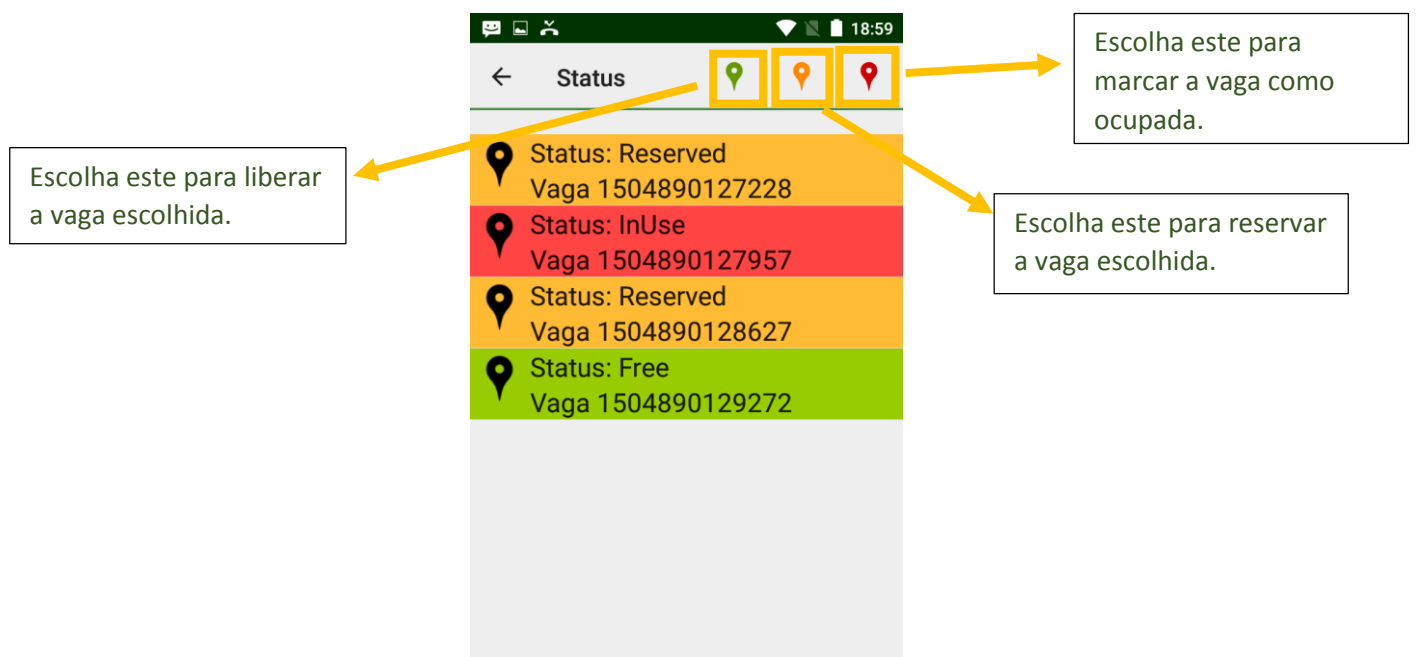


Figura 11 – Menu de opções do status da vaga

Ao escolher a opção de status de interesse a cor da vaga selecionada muda de acordo com o novo status escolhido. Ao tentar mudar o status de uma vaga reservada a mensagem da Figura 12 aparecerá.

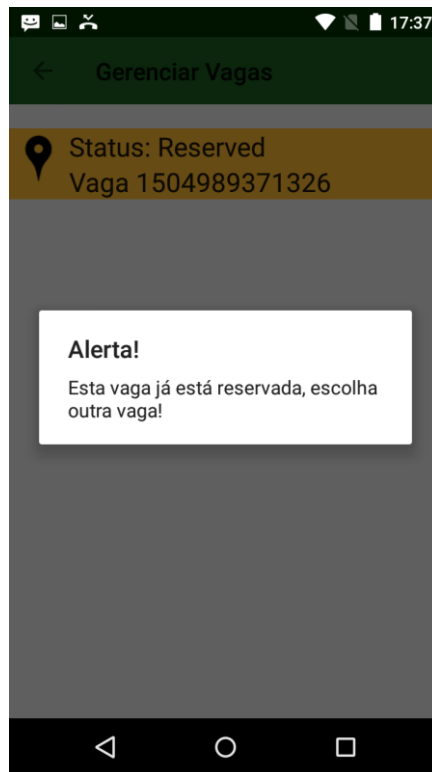


Figura 12 – Mensagem de alerta

Ao escolher a opção “Rota” a rota do local atual do usuário até o estacionamento escolhido será traçada e à medida que o motorista dirige até o local sua localização vai sendo atualizada.

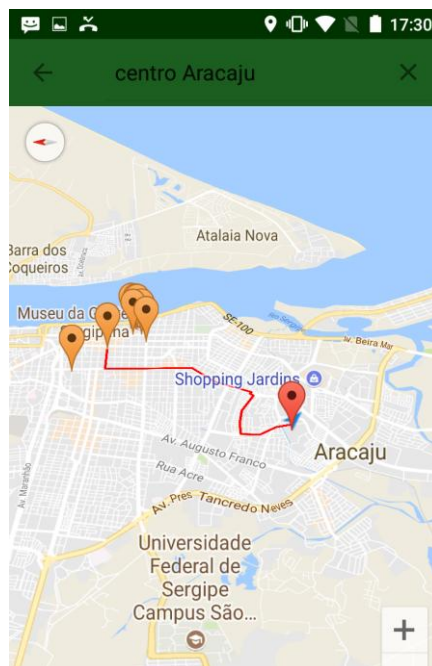


Figura 13 – Rota do local atual do usuário até o estacionamento escolhido

Ao clicar no ícone que fica na parte superior esquerda um menu lateral aparecerá com todas as opções disponíveis no aplicativo.

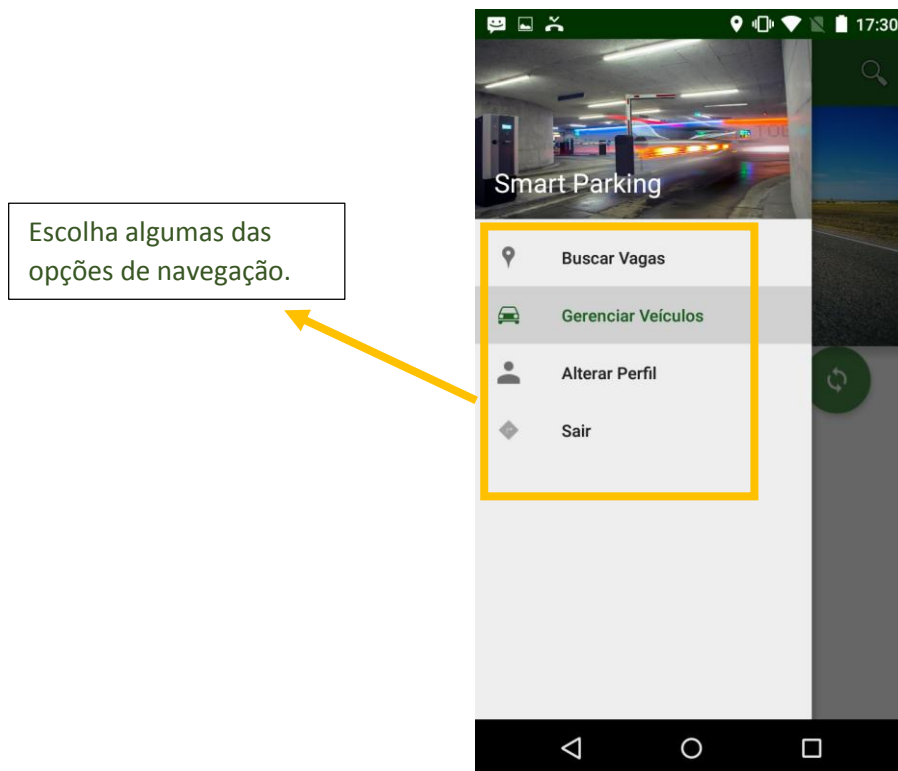


Figura 15 – Menu lateral

5. Gerenciamento de Veículos

Ao escolher a opção Gerenciar Veículos, o usuário será direcionado a tela com as informações do veículo utilizado atualmente. Caso não exista nenhum veículo cadastrado uma imagem informando que não existem dados cadastrados aparecerá.

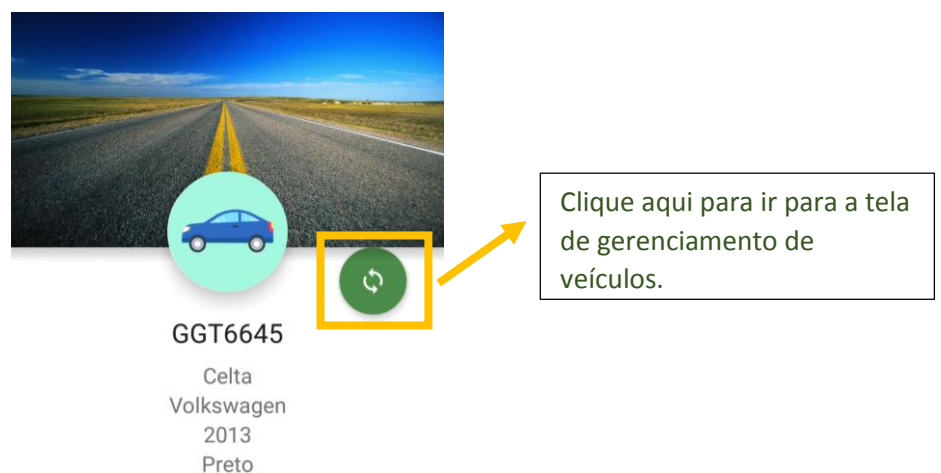


Figura 16 – Tela de informações do veículo atual

Ao clicar no botão flutuante o usuário será redirecionado a tela de gerenciamento de veículos na qual é apresentada uma lista com todos os veículos cadastrados.

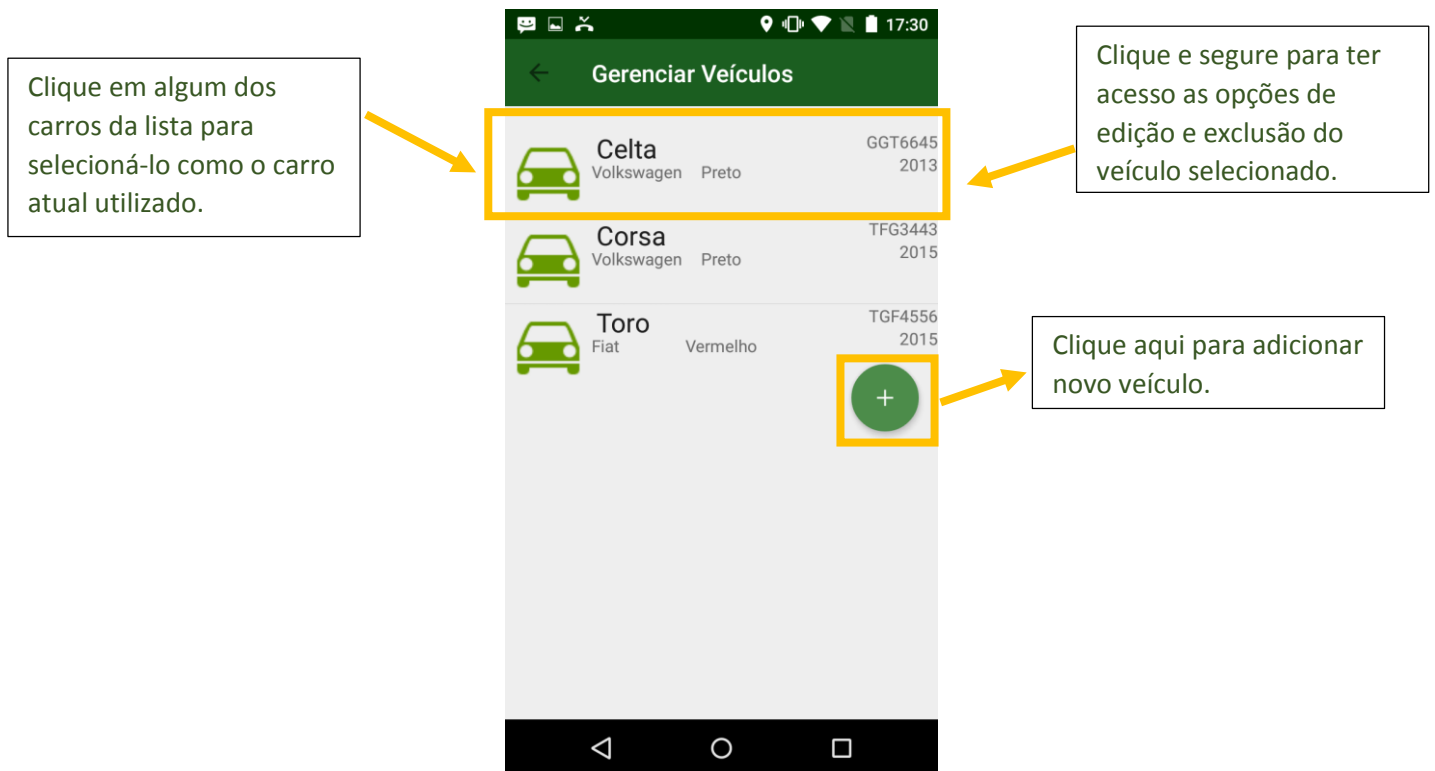


Figura 17 – Tela de Gerenciamento de Veículos

Ao clicar no botão flutuante com o símbolo de '+' (mais) uma caixa de diálogo aparecerá para que o usuário insira as informações do veículo.

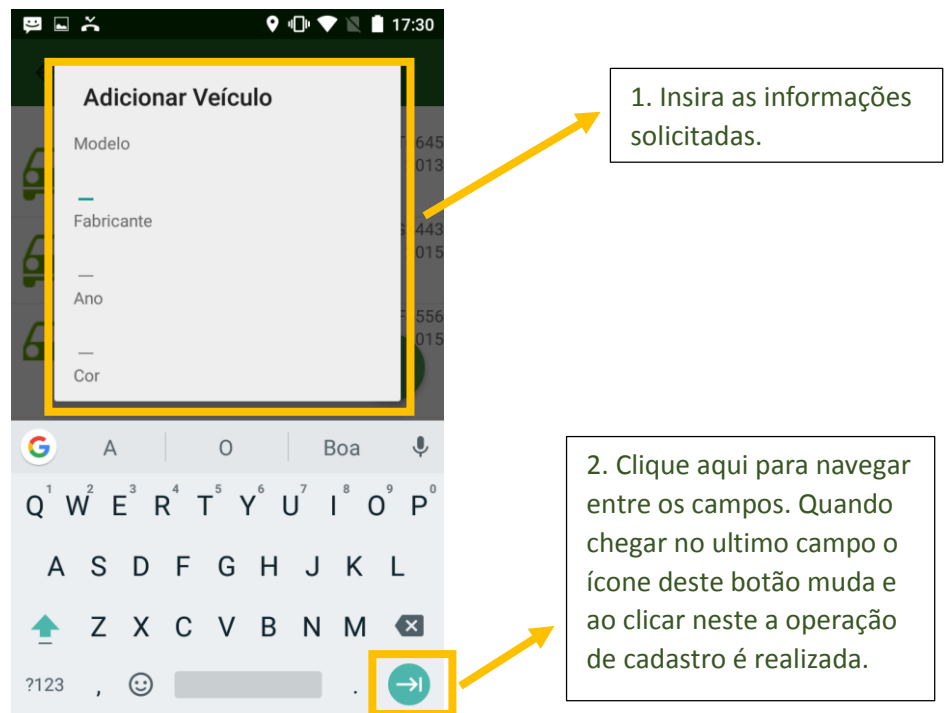


Figura 18 – Caixa de diálogo para cadastro de novo veículo

Enquanto insere as informações o usuário navega pelos campos através do próprio teclado e ao finalizar o botão do teclado mudará para o modo “Done” no qual ao clicar a operação é realizada.

Ao clicar e segurar em algum dos veículos da lista um menu de opções aparecerá para o usuário, através de deste o usuário pode editar as informações daquele veículo ou excluí-lo.

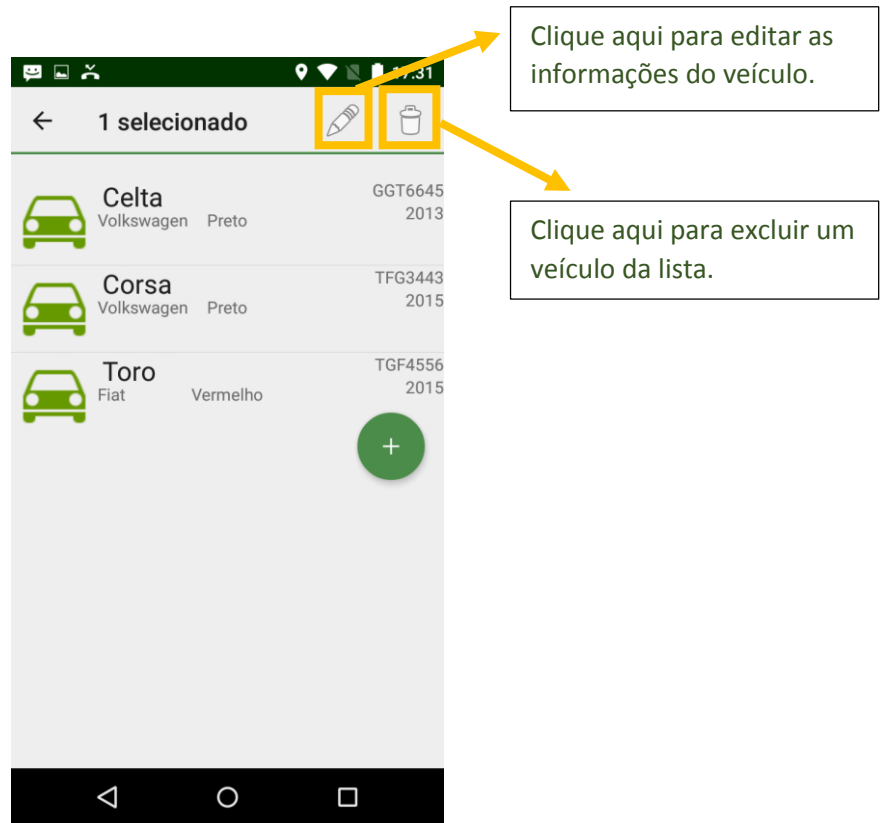


Figura 19 – Menu de opções do veículo

6. Atualizar Perfil

Para atualizar as informações do perfil, no menu lateral escolha a opção atualizar perfil, uma tela com as informações do usuário atual aparecerá.

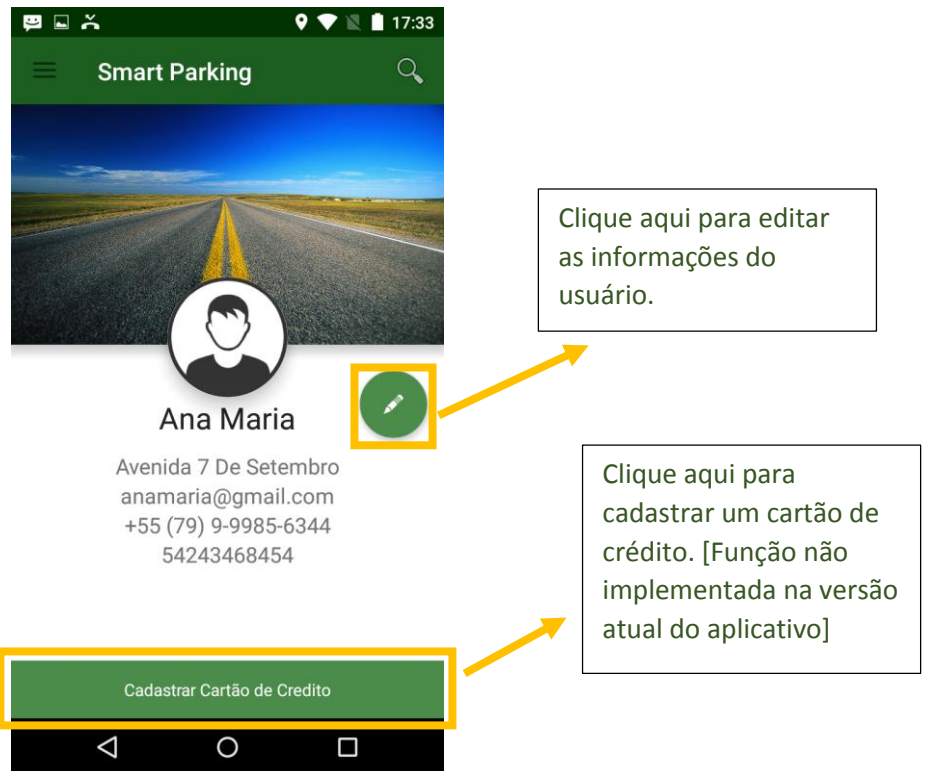


Figura 20 – Tela de informações do usuário

Para editar as informações do usuário, este deve clicar no botão flutuante com o ícone do lápis. O usuário será redirecionado para a tela de edição das informações (Figura 21).

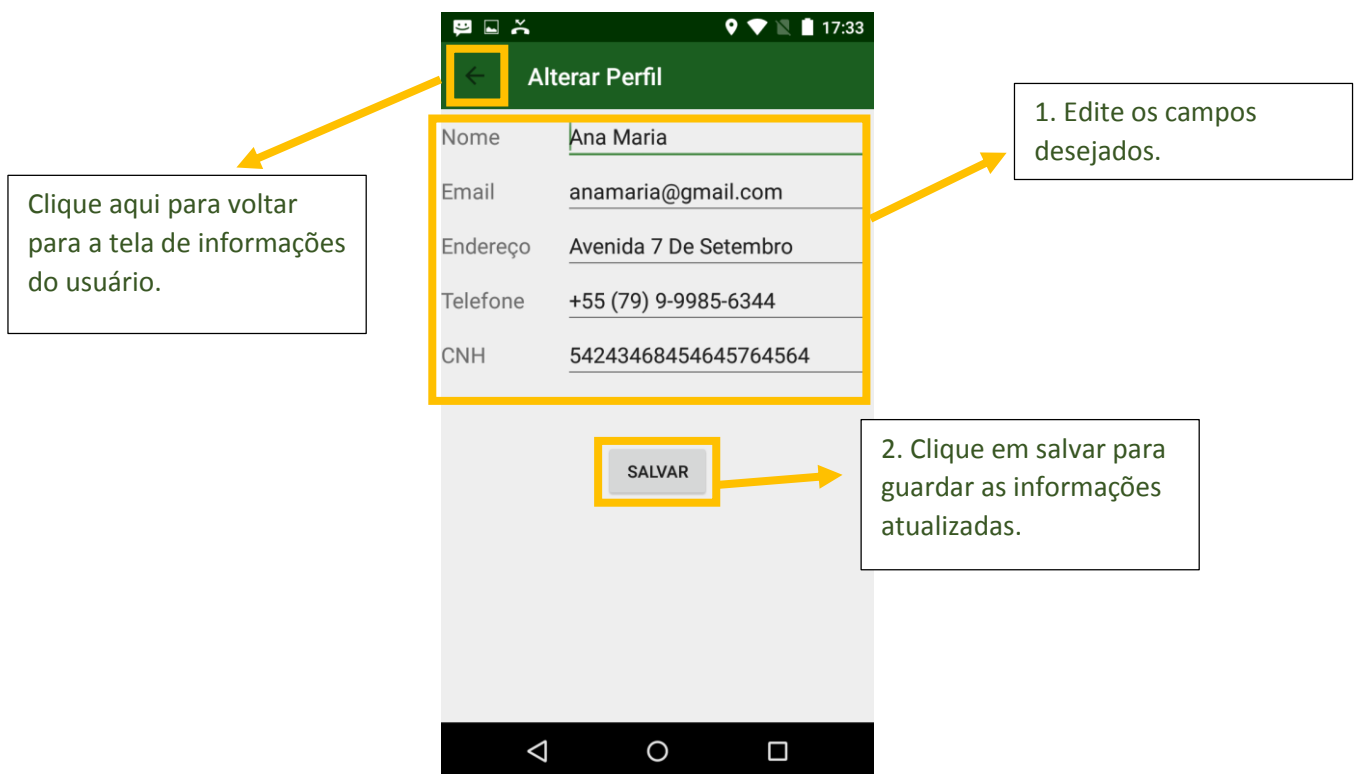


Figura 21 – Tela de edição das informações do usuário

7. Tela de Gerenciamento do Estacionamento

Ao se autenticar no sistema com uma conta do tipo Gestor de Estacionamento a tela inicial que será exibida será uma com as informações do estacionamento que está sendo gerenciado atualmente.

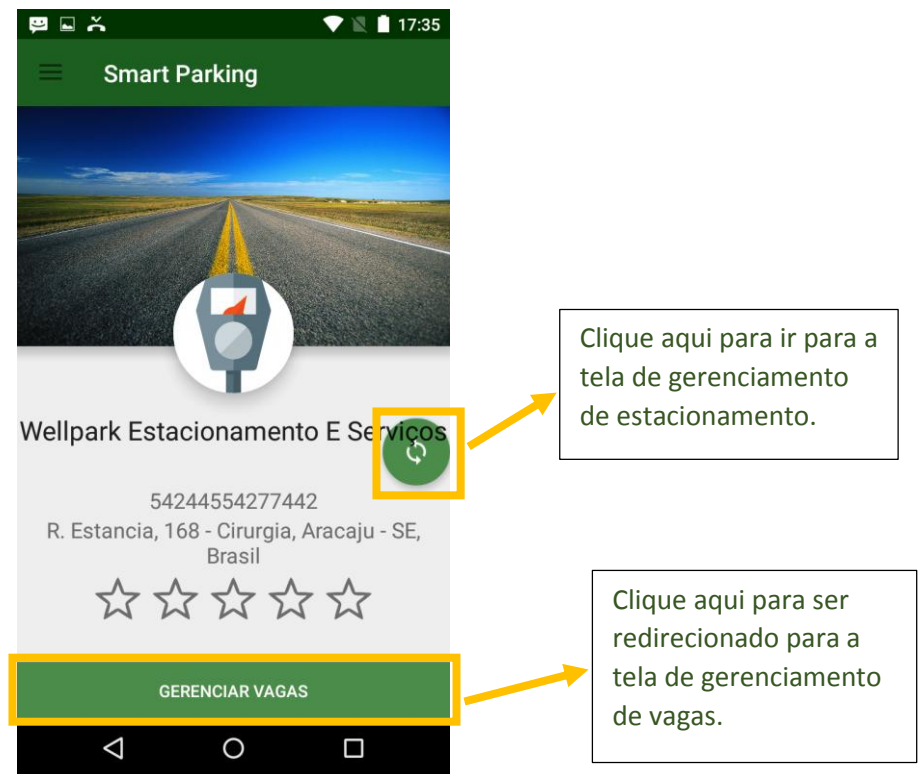


Figura 22 – Tela inicial do Gestor de Estacionamento

Ao clicar no botão flutuante o usuário será redirecionado para a tela de gerenciamento dos estacionamentos, nesta tela será exibida uma lista com todos os estacionamentos cadastrados.

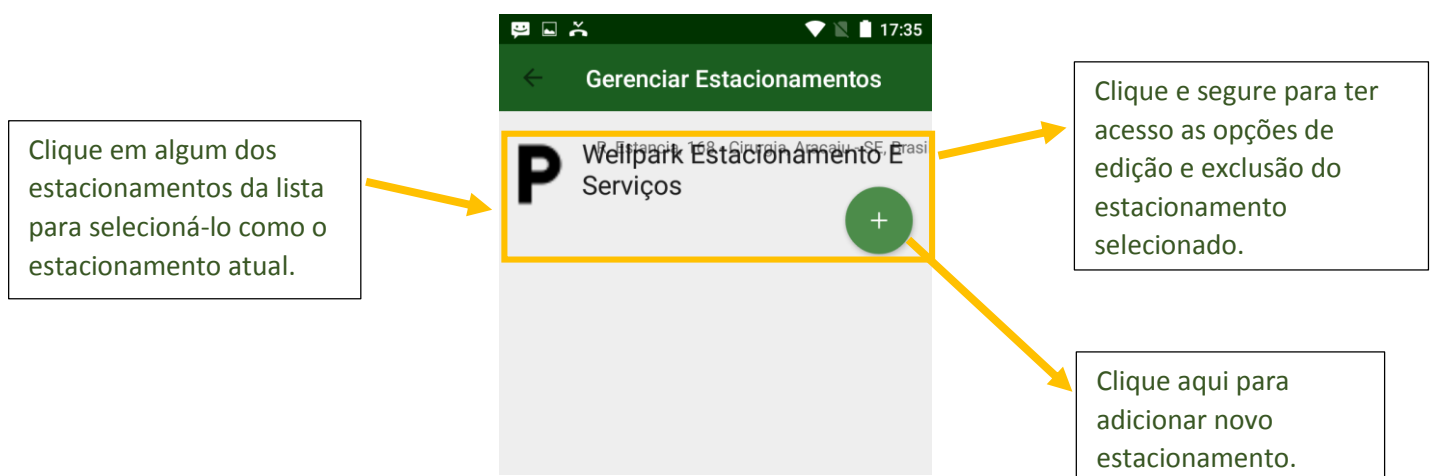


Figura 23 – Tela inicial do Gestor de Estacionamento

Ao clicar no botão flutuante com o símbolo de ‘+’ (mais) uma caixa de diálogo aparecerá para que o usuário insira as informações do estacionamento.

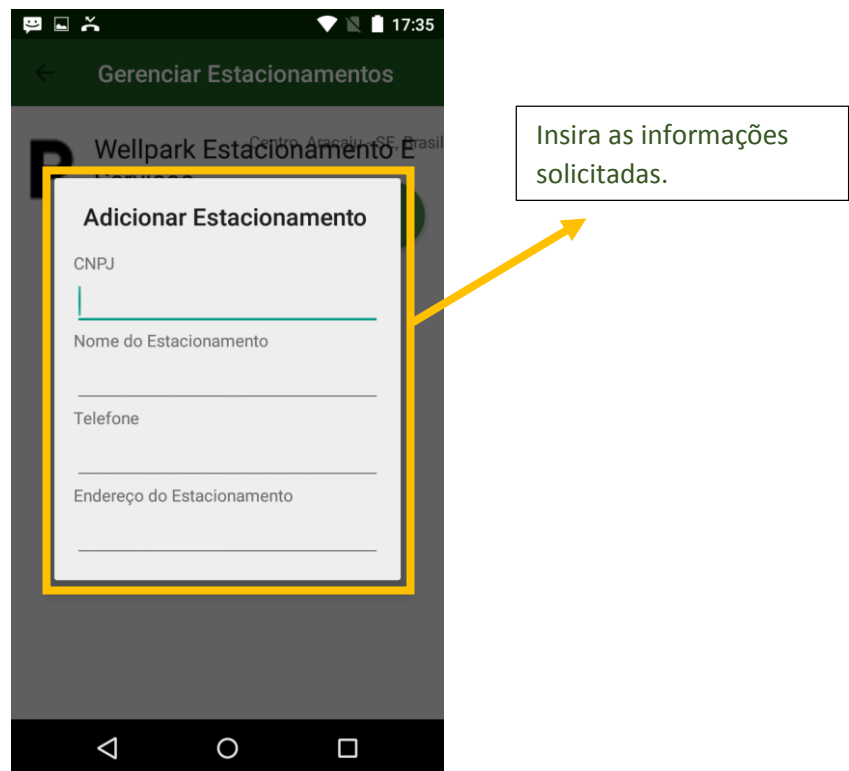


Figura 23 – Caixa de diálogo para cadastro de novo estacionamento

Enquanto insere as informações o usuário navega pelos campos através do próprio teclado e ao finalizar o botão do teclado mudará para o modo “Done” no qual ao clicar a operação é realizada.

Ao clicar e segurar em algum dos estacionamentos da lista um menu de opções aparecerá para o usuário, através de deste o usuário pode editar as informações daquele estacionamento, excluí-lo ou cadastrar vagas.

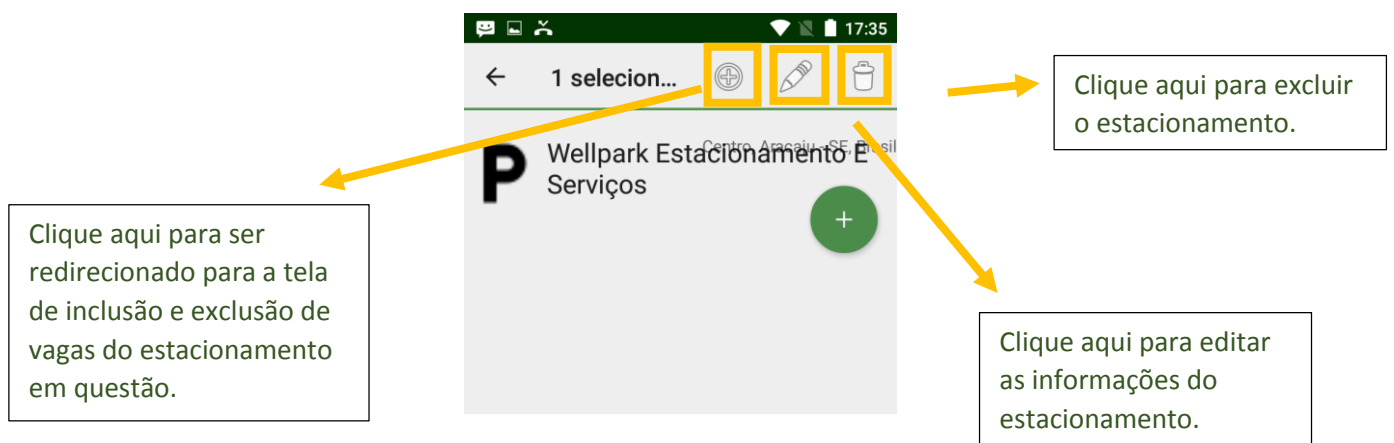


Figura 24 – Menu de opções do estacionamento

Ao clicar na opção de cadastrar vagas (símbolo de mais dentro de um círculo) o usuário será redirecionado para a tela de inclusão e exclusão de vagas apresentada na Figura 25.

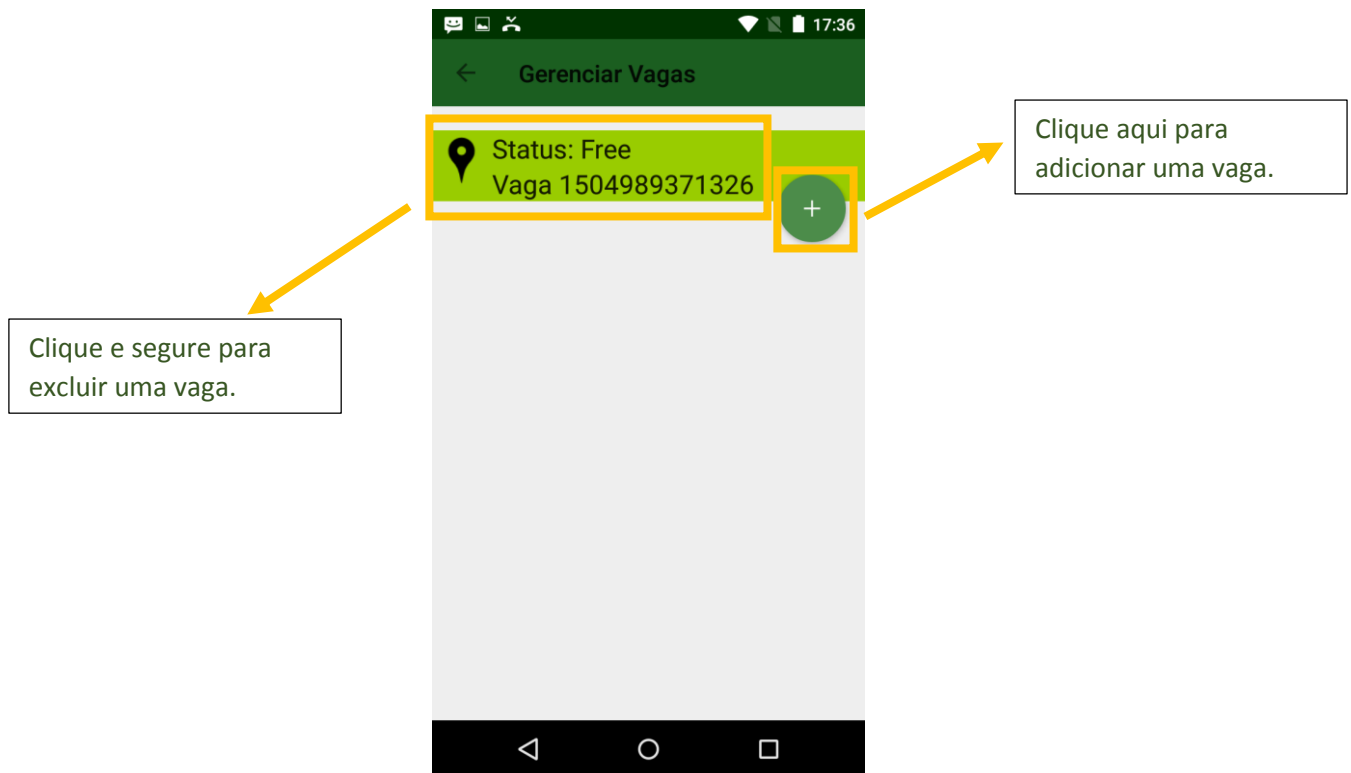


Figura 25 – Tela de inclusão e exclusão de vagas

Ao clicar em Gerenciar Vagas (Figura 22), o usuário será redirecionado para a mesma tela apresentada na Figura 11 e é capaz de executar as mesmas operações que apresentado na seção 4.

8. Gerenciamento de reservas

Toda vez que um usuário do tio motorista reservar uma vaga de um respectivo estacionamento, está solicitação fica registrada em uma lista na tela de gerenciamento de reservas do usuário gestor do estacionamento em questão. Para acessar está tela o usuário deve abrir o menu lateral e escolher a opção Gerenciar Reservas.

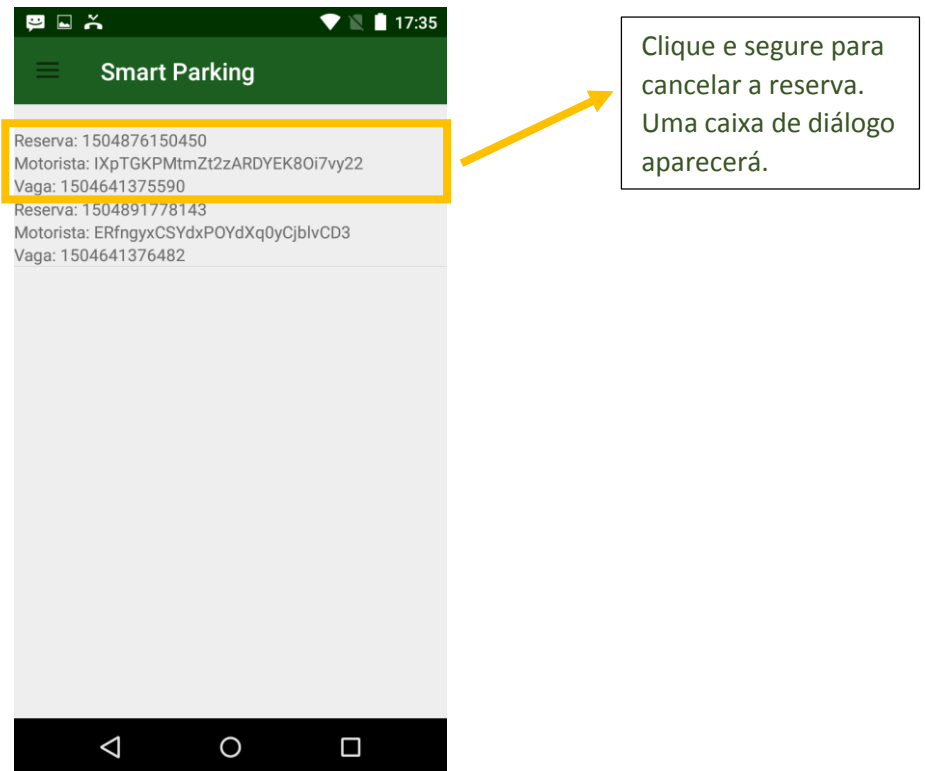


Figura 26 – Tela de Gerenciamento de Reservas

Atualmente na tela de reservas é apenas possível cancelar a reserva se desejado. Para isso clique e segure na reserva que deseja cancelar, uma mensagem aparecerá perguntando se deseja cancelar a reserva.

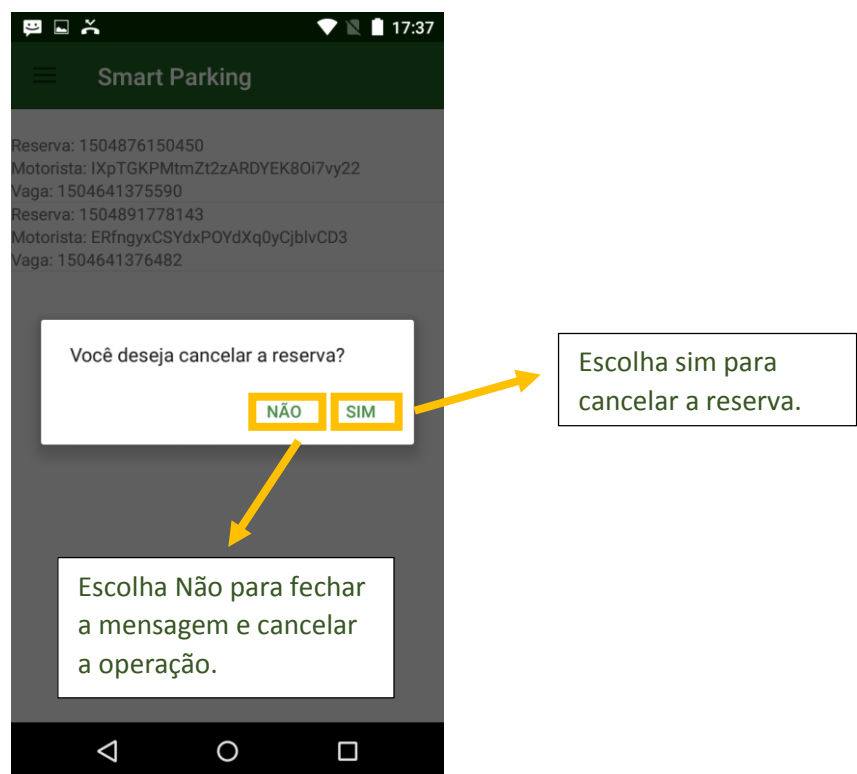


Figura 27 – Caixa de diálogo com mensagem de confirmação

Ao escolher a opção “Sim” a reserva é retirada da lista e a respectiva vaga tem o seu status mudado para livre. Caso a opção “Não” seja escolhida a mensagem desaparece e nenhuma ação é executada.

Para atualização do perfil do usuário do tipo Gestor do estacionamento, os passos que devem ser seguidos são os mesmos apresentados na seção 6.

Para sair do aplicativo escolha a opção “Sair” no menu lateral (Figura 15).

SMART PARKING

Desenvolvedores

Samila Ruane Barboza Santos

Sílvia Rodrigo Lima Passos

Contato

samilaruane@gmail.com

silviosrlp@gmail.com